

Let's Build a
COMPOSER PACKAGE

Ben Ramsey
php[tek] • April 23, 2024



WELCOME

A bit about me

- Senior Staff Software Engineer at Intuit (working on Mailchimp API)
- PHP 8.1 and 8.2 release manager
- Author of many packages...

- ramsey/uuid
- league/oauth2-client
- skillshare/formatphp
- ramsey/collection
- ramsey/composer-repl
- ramsey/conventional-commits

- ramsey/devtools
- ramsey/http-range
- ramsey/php-library-starter-kit
- ramsey/pygments
- and more!

WHO IS THIS FOR?

- ***Individuals*** who want to open source a library.
- ***Companies*** who want to open source a library.
- ***Others*** who just want to learn how it all works.

WHAT WILL WE COVER?

- What makes a *good* or *high quality* Composer package?
- Common *patterns* of packages.
- How to *publish* a package.
- *Maintaining* a package and building a *community*.

This is hands-on.

(if you want it to be)

ASSUMPTIONS

And caveats

- You'll need to install:
 - PHP 8.1+ — www.php.net/install
 - Composer — getcomposer.org/download
 - Git — git-scm.com/downloads

INSTALLING

On macOS or Linux with Homebrew

- Go to brew.sh to download and install Homebrew.
- Install PHP, Composer, and Git with:

```
brew install php composer git
```

INSTALLING

On Windows with Chocolatey

- Go to chocolatey.org/install to download and install Chocolatey.
- Install PHP, Composer, and Git with:

```
choco install php composer git
```

INSTALLING

On Debian Bookworm or Ubuntu Jammy (or later)

- Install using:

```
apt install php composer git
```

- If you get a PHP version earlier than 8.1, use Ondřej Surý's packages:
deb.sury.org.

INSTALLING

On Fedora 36 (or later)

- Install using:

```
yum install php composer git
```

- If you get a PHP version earlier than 8.1, use Remi Collet's packages:
rpms.remirepo.net.

INSTALLING

On on anything else

- Other installation methods exist.
- Check the documentation for your install method of choice.



ASSUMPTIONS

And caveats

- You'll need accounts at:
 - GitHub — github.com/signup
 - Packagist — packagist.org/login/github
 - Codecov — codecov.io/login/gh



That's just, like.... your opinion, man.

ASSUMPTIONS

And caveats

- This is all highly opinionated.
- For example, I'm using:
 - PHP Package Development Standards (PHP-PDS)
 - PHPUnit
 - PHP_CodeSniffer
 - PHPStan

ASSUMPTIONS

And caveats

- The tools I'm using aren't necessarily requirements for creating packages.
 - *Well, PHP and Composer are.* 😊
- You could use Mercurial, Bitbucket, Travis CI, and Coveralls, for example.
- Or Pest, phpcs-fixer, Phan, etc.
- This tutorial doesn't cover these tools.

LET'S BRAINSTORM

What does our package do?

What does our package do?

“Hello, World!”

What's our package namespace?

What's our package namespace?

```
WildGarlic\HelloWorld
```

What's our package name?

What's our package name?

wild-garlic/hello-world

WHAT IS A PACKAGE?

A PACKAGE...

- Is a collection of code.
- Addresses a single problem or problem domain.[†]
- Provides a simple interface to underlying functionality.[†]
- Is often highly abstracted.[†]
- Is distributed through a package manager (i.e., npm, Cargo, pip, RubyGems, Composer, etc.).[†]

† *Usually*

PACKAGE REQUIREMENTS

The bare minimum

- `composer.json`
- Some code...maybe.

PACKAGE REQUIREMENTS

The bare minimum: composer.json

```
{
    "name": "wild-garlic/hello-world",
    "description": "A basic Hello, World example.",
    "type": "library",
    "autoload": {
        "psr-4": {
            "WildGarlic\\HelloWorld\\": "./"
        }
    }
}
```

PACKAGE REQUIREMENTS

The bare minimum: HelloWorld.php

```
<?php declare(strict_types=1);  
  
namespace WildGarlic\HelloWorld;  
  
class HelloWorld  
{  
    public function greet(string $name = 'World'): string  
    {  
        return "Hello, $name!";  
    }  
}
```

Search packages...

wild-garlic/hello-world

composer require wild-garlic/hello-world

A basic "Hello, World" example.

Maintainers



Details

github.com/wild-garlic/hello-world

[Source](#)

[Issues](#)

- Installs: 2
- Dependents: 0
- Suggesters: 0
- Security: 0
- Stars: 0
- Watchers: 1
- Forks: 0
- Open Issues: 5

dev-phptek2024-minimal 2023-10-29 19:28 UTC

requires	requires (dev)	suggests
None	None	None
provides	conflicts	replaces
None	None	None

dev-main

0.1.0

- [dev-dependabot/github_actions/ri...](#)
- [dev-dependabot/github_actions/gi...](#)
- [dev-dependabot/github_actions/s...](#)
- [dev-longhorn2023](#)
- [dev-dependabot/github_actions/c...](#)

PACKAGE REQUIREMENTS

The bare minimum: We can require and use it

```
composer require wild-garlic/hello-world:dev-phptek2024-minimal
```

PACKAGE REQUIREMENTS

The bare minimum: We can require and use it

```
<?php declare(strict_types=1);  
  
namespace App;  
  
use WildGarlic\HelloWorld\HelloWorld;  
  
require_once 'vendor/autoload.php';  
  
$hello = new HelloWorld();  
  
echo $hello->greet('php[tek]');
```

That's all, folks!

(just kidding)

PACKAGE REQUIREMENTS

The bare minimum: Links to examples

- **phptek2024-minimal** branch on GitHub:
 - github.com/wild-garlic/hello-world/tree/phptek2024-minimal
- **dev-phptek2024-minimal** version on Packagist:
 - packagist.org/packages/wild-garlic/hello-world#dev-phptek2024-minimal

ANATOMY OF A PACKAGE

Does directory structure matter?

“Users spend most of their time on other sites.”

Jakob Nielsen, “End of Web Design,” July 22, 2000 (a.k.a. *Jakob's Law of Internet User Experience*)

*This applies to developer
experience, too.*

DIRECTORY STRUCTURE

PHP Package Development Standards (PHP-PDS)

- Research-based.
- Evaluated thousands of Composer packages across the PHP ecosystem.
- Analyzed the results to derive the most common directory structure.
- php-pds.com

<code>bin/</code>	command-line executables
<code>config/</code>	configuration files
<code>docs/</code>	project documentation
<code>public/</code>	web server files (rarely used in packages)
<code>resources/</code>	other resources, i.e., data files, etc.
<code>src/</code>	PHP source code
<code>tests/</code>	test code
???	Other root-level directories are okay (i.e., <code>tools/</code> , etc.)

DIRECTORY STRUCTURE

Discussion

- The earlier minimal example placed `HelloWorld.php` in the package root.
- Some projects, like `Symfony`, follow this pattern for their components.
- I'll assume `src/` for the rest of my examples.
- Most packages will have `src/` and `tests/` at a minimum.
- Many also include `docs/`.
- Some include `bin/`, if providing command line tools.

IMPORTANT FILES

README

- README provides the most important information users need to know.
 - Installation
 - Usage
- Depending on the type of content, it might be named README.md, README.markdown, README.txt, README.rst, etc.

IMPORTANT FILES

CHANGELOG

- CHANGELOG provides a list of changes for each version of your package, including any backward-compatibility (BC) breaks.
- Helps users understand new features, fixes, and potential issues they might encounter when upgrading.
- Meant for humans.
- No industry-standard format, though Keep A Changelog is popular.



keep a changelog

Don't let your friends dump git logs into changelogs.

Version **1.1.0**

```
# Changelog

All notable changes to this project will be documented in this file.

The format is based on [Keep a Changelog](https://keepachangelog.com/en/1.0.0/),
and this project adheres to [Semantic Versioning](https://semver.org/spec/v2.0.0.html).

## [Unreleased]

## [1.1.1] - 2023-03-05

### Added

- Arabic translation (#444).
- v1.1 French translation.
```

What is a changelog?

A changelog is a file which contains a curated, chronologically ordered list of notable changes for each version of a project.

Why keep a changelog?

3.5.0 - 2023-04-27

Added

- Add an option for version 2.1 of the Contributor Covenant
- Update GitHub workflows to support auto-merging of Dependabot pull requests

Changed

- Update ramsey/devtools to version 2.0
- Increase minimum PHP version to 8.1

Deprecated

- Nothing.

Removed

- Nothing.

Fixed

- Stop passing `starter-kit` command name to avoid confusing newer versions of symfony/console.

Security

- Nothing.

IMPORTANT FILES

LICENSE

- What permissions do your users have?
- Two primary styles of open source licenses:
 - Permissive
 - Copyleft
- I am not a lawyer; this is not legal advice.
- Choose an **OSI-approved** or **FSF-approved** open source license.



Home / Licenses

OSI Approved Licenses

Open source licenses are licenses that comply with the Open Source Definition – in brief, they allow software to be freely used, modified, and shared. To be approved by the Open Source Initiative (also known as the OSI) a license must go through the Open Source Initiative's [license review process](#).



SEARCH LICENSES ▾

Search for:

CATEGORIES ▾

- INTERNATIONAL
- NON-REUSABLE
- OTHER/MISCELLANEOUS
- POPULAR / STRONG COMMUNITY
- REDUNDANT WITH MORE POPULAR
- SPECIAL PURPOSE
- SUPERSEDED
- UNCATEGORIZED
- VOLUNTARILY RETIRED

Free Software Supporter: [Sign up](#)

[JOIN THE FSF](#)



Various Licenses and Comments about Them

Introduction

Software Licenses

- [GPL-Compatible Free Software Licenses](#)
- [GPL-Incompatible Free Software Licenses](#)
- [Nonfree Software Licenses](#)

Licenses For Documentation

- [Free Documentation Licenses](#)
- [Nonfree Documentation Licenses](#)

Licenses for Other Works

- [Licenses for Works of Practical Use besides Software and Documentation](#)
- [Licenses for Fonts](#)
- [Licenses for Works stating a Viewpoint \(e.g., Opinion or Testimony\)](#)
- [Licenses for Designs for Physical Objects](#)

If a package does not specify a license, it is not free software!

IMPORTANT FILES

Other files?

Yes. We'll get to them in a bit.

CREATE OUR PACKAGE

PHP LIBRARY STARTER KIT

A starting point

- A starter kit for setting up a new PHP library package.
- Highly opinionated (built for me)
 - PHPUnit for tests
 - PHP_CodeSniffer for coding standards (using ramsey/coding-standard)
 - PHPStan and Psalm for static analysis
 - CaptainHook to manage Git hooks

Let's create a package!

- We'll use `composer create-project`.
- Follow along, if you like.

```
composer create-project ramsey/php-library-starter-kit hello-world
```



ramsey@Celebrian:~/Developer/Repos/wild-garlic

Developer/Repos/wild-garlic

> _



php composer create-project ramsey/php-library-starter-kit hello-world

```
'commit-msg' hook installed successfully
'pre-push' hook installed successfully
'pre-commit' hook installed successfully
'prepare-commit-msg' hook installed successfully
'post-commit' hook installed successfully
'post-merge' hook installed successfully
'post-checkout' hook installed successfully
'post-rewrite' hook installed successfully
No security vulnerability advisories found
> Ramsey\Dev\LibraryStarterKit\Wizard::start
```

Welcome to the PHP Library Starter Kit!

=====

I'll ask you a series of questions about the library you're creating. When finished, I'll set up a repository with an initial set of files that you may customize to suit your needs.

Are you ready to get started? (yes/no) [no]:

> _



php composer create-project ramsey/php-library-starter-kit hello-world

```
'prepare-commit-msg' hook installed successfully  
'post-commit' hook installed successfully  
'post-merge' hook installed successfully  
'post-checkout' hook installed successfully  
'post-rewrite' hook installed successfully  
No security vulnerability advisories found  
> Ramsey\Dev\LibraryStarterKit\Wizard::start
```

Welcome to the PHP Library Starter Kit!

=====

I'll ask you a series of questions about the library you're creating. When finished, I'll set up a repository with an initial set of files that you may customize to suit your needs.

Are you ready to get started? (yes/no) [no]:

> yes

What is the GitHub username or org name for your package?:

> _



php composer create-project ramsey/php-library-starter-kit hello-world

'post-checkout' hook installed successfully

'post-rewrite' hook installed successfully

No security vulnerability advisories found

> Ramsey\Dev\LibraryStarterKit\Wizard::start

Welcome to the PHP Library Starter Kit!

=====

I'll ask you a series of questions about the library you're creating. When finished, I'll set up a repository with an initial set of files that you may customize to suit your needs.

Are you ready to get started? (yes/no) [no]:

> yes

What is the GitHub username or org name for your package?:

> wild-garlic

What is your package vendor name? [wild-garlic]:

> _



I'll ask you a series of questions about the library you're creating. When finished, I'll set up a repository with an initial set of files that you may customize to suit your needs.

Are you ready to get started? (yes/no) [no]:

> yes

What is the GitHub username or org name for your package?:

> wild-garlic

What is your package vendor name? [wild-garlic]:

>

What is your package name? [wild-garlic/hello-world]:

>

Enter a brief description of your library:

> _



finished, I'll set up a repository with an initial set of files that you may customize to suit your needs.

Are you ready to get started? (yes/no) [no]:

> yes

What is the GitHub username or org name for your package?:

> wild-garlic

What is your package vendor name? [wild-garlic]:

>

What is your package name? [wild-garlic/hello-world]:

>

Enter a brief description of your library:

> A basic "Hello, World" example.

What is your name? [Wild Garlic]:

> _



php composer create-project ramsey/php-library-starter-kit hello-world

What is the GitHub username or org name for your package?:

> wild-garlic

What is your package vendor name? [wild-garlic]:

>

What is your package name? [wild-garlic/hello-world]:

>

Enter a brief description of your library:

> A basic "Hello, World" example.

What is your name? [Wild Garlic]:

>

What is your email address? [149326471+wild-garlic@users.noreply.github.com]:

>

What is your website address?:

> _



php composer create-project ramsey/php-library-starter-kit hello-world

What is your package vendor name? [wild-garlic]:

>

What is your package name? [wild-garlic/hello-world]:

>

Enter a brief description of your library:

> A basic "Hello, World" example.

What is your name? [Wild Garlic]:

>

What is your email address? [149326471+wild-garlic@users.noreply.github.com]:

>

What is your website address?:

> https://wildgarlic.example.com

Are you the copyright holder? (yes/no) [yes]:

> _



ramsey@Celebrian:~/Developer/Repos/wild-garlic

What is your email address? [149326471+wild-garlic@users.noreply.github.com]:
>

What is your website address?:
> https://wildgarlic.example.com

Are you the copyright holder? (yes/no) [yes]:
> ^C

I've saved your progress. When you're ready to return to the starter kit wizard, enter:

```
cd /Users/ramsey/Developer/Repos/wild-garlic/hello-world  
composer starter-kit
```

Developer/Repos/wild-garlic took 2m

>
_

```
{
  "authorEmail": "149326471+wild-garlic@users.noreply.github.com",
  "author Holds Copyright": true,
  "authorName": "Wild Garlic",
  "authorUrl": "https://wildgarlic.example.com",
  "codeOfConduct": "None",
  "codeOfConductCommittee": null,
  "codeOfConductEmail": null,
  "codeOfConductPoliciesUrl": null,
  "codeOfConductReportingUrl": null,
  "copyrightEmail": null,
  "copyrightHolder": null,
  "copyrightUrl": null,
  "copyrightYear": null,
  "githubUsername": "wild-garlic",
  "license": "Proprietary",
  "packageDescription": "A basic \"Hello, World\" example.",
  "packageKeywords": [],
```

NORMAL

↶ main

← kit-answers

jso...

utf-8[!EOL][unix]

33% L:9/27 ≡ %:1

```
"hello-world/.starter-kit-answers" [noeol] 27L, 909B
```



> https://ben.ramsey.dev

What is the copyright year? [2023]:

>

Choose a license for your project [Proprietary]:

- [1] Proprietary
- [2] Apache License 2.0
- [3] BSD 2-Clause "Simplified" License
- [4] BSD 3-Clause "New" or "Revised" License
- [5] Creative Commons Zero v1.0 Universal
- [6] GNU Affero General Public License v3.0 or later
- [7] GNU General Public License v3.0 or later
- [8] GNU Lesser General Public License v3.0 or later
- [9] Hippocratic License 2.1
- [10] MIT License
- [11] MIT No Attribution
- [12] Mozilla Public License 2.0
- [13] Unlicense

> _



>

Choose a license for your project [Proprietary]:

- [1] Proprietary
- [2] Apache License 2.0
- [3] BSD 2-Clause "Simplified" License
- [4] BSD 3-Clause "New" or "Revised" License
- [5] Creative Commons Zero v1.0 Universal
- [6] GNU Affero General Public License v3.0 or later
- [7] GNU General Public License v3.0 or later
- [8] GNU Lesser General Public License v3.0 or later
- [9] Hippocratic License 2.1
- [10] MIT License
- [11] MIT No Attribution
- [12] Mozilla Public License 2.0
- [13] Unlicense

> 10

Do you want to include a security policy? (yes/no) [yes]:

> _



[12] Mozilla Public License 2.0

[13] Unlicense

> 10

Do you want to include a security policy? (yes/no) [yes]:

> yes

What email address should researchers use to submit vulnerability reports?:

> security@ramsey.dev

At what URL should researchers submit vulnerability reports?:

>

Choose a code of conduct for your project [None]:

[1] None

[2] Contributor Covenant Code of Conduct, version 1.4

[3] Contributor Covenant Code of Conduct, version 2.0

[4] Contributor Covenant Code of Conduct, version 2.1

[5] Citizen Code of Conduct, version 2.3

>

_



What email address should researchers use to submit vulnerability reports?:

> security@ramsey.dev

At what URL should researchers submit vulnerability reports?:

>

Choose a code of conduct for your project [None]:

[1] None

[2] Contributor Covenant Code of Conduct, version 1.4

[3] Contributor Covenant Code of Conduct, version 2.0

[4] Contributor Covenant Code of Conduct, version 2.1

[5] Citizen Code of Conduct, version 2.3

> 4

What email address should people use to report code of conduct issues?:

> ben@ramsey.dev

Enter a set of comma-separated keywords describing your library:

> _



At what URL should researchers submit vulnerability reports?:

>

Choose a code of conduct for your project [None]:

[1] None

[2] Contributor Covenant Code of Conduct, version 1.4

[3] Contributor Covenant Code of Conduct, version 2.0

[4] Contributor Covenant Code of Conduct, version 2.1

[5] Citizen Code of Conduct, version 2.3

> 4

What email address should people use to report code of conduct issues?:

> ben@ramsey.dev

Enter a set of comma-separated keywords describing your library:

> example,sample,hello,world,composer,package

What is the library's root namespace? [Wild\Garlic\Hello\World]:

> _



ramsey@Celebrian:~/Developer/Repos/wild-garlic/hello-world

Runtime: PHP 8.2.10

Configuration: /Users/ramsey/Developer/Repos/wild-garlic/hello-world/phpunit.xml
.dist

. 1 / 1 (100%)

Time: 00:00.007, Memory: 10.00 MB

OK (1 test, 1 assertion)

[OK] Congratulations! Your project, wild-garlic/hello-world, is ready!

Your project is available at
/Users/ramsey/Developer/Repos/wild-garlic/hello-world.

hello-world on ↗ main via 🐘 v8.2.10 took 6m37s

>

_



ramsey@Celebrian:~/Developer/Repos/wild-garlic/hello-world

hello-world on [↩ main](#) via  v8.2.10

> ls

CHANGELOG.md	bin/	conventional-commits.json	resources/
CODE_OF_CONDUCT.md	build/	docs/	src/
CONTRIBUTING.md	captainhook.json	phpcs.xml.dist	tests/
LICENSE	codecov.yml	phpstan.neon.dist	vendor/
README.md	composer.json	phpunit.xml.dist	
SECURITY.md	composer.lock	psalm.xml	

hello-world on [↩ main](#) via  v8.2.10

> _

PUBLISH OUR PACKAGE

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * **Repository name ***

wild-garlic / hello-worl

hello-worl is available.

Great repository names are short and memorable. Need inspiration? How about [fictional-guide](#) ?

Description (optional)

A basic "Hello, World" example.

- Public**
Anyone on the internet can see this repository. You choose who can commit.
- Private**
You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)



ramsey@Celebrian:~/Developer/Repos/wild-garlic/hello-world

hello-world on  main via  v8.2.10

> _

hello-world Public Pin Unwatch (1) Fork (0) Star (0)

main 6 branches 0 tags Go to file Add file Code

wild-garlic chore: initialize project using ramsey/php-library-starter-kit ✓ f0854ac 8 minutes ago 1 commit		
.github	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
bin	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
build	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
docs	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
resources	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
src	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
tests	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
.editorconfig	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
.gitattributes	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
.gitignore	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
CHANGELOG.md	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
CODE_OF_CONDUCT.md	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
CONTRIBUTING.md	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
LICENSE	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago
README.md	chore: initialize project using ramsey/php-library-starter-kit	8 minutes ago

About

A basic "Hello, World" example.
Readme
MIT license
Activity
0 stars
1 watching
0 forks

Releases

No releases published
[Create a new release](#)

Languages



PUBLISH OUR PACKAGE

Review: Push to GitHub

- Create repository at github.com/new.
- Connect your local repository to the remote one:

```
git remote add origin git@github.com:yourname/repository.git
```
- Push the main branch to the remote repository:

```
git push -u origin main
```
- View your repository on GitHub!

packagist.org/packages/submit

Packagist *The PHP Package Repository* Browse Submit wild-garlic

Search packages...

Submit package

Repository URL (Git/Svn/Hg)

Check

Trying to share private code?
Use [Private Packagist](#) to share code through Composer without publishing it for everyone on Packagist.org.

Please make sure you have read the package [naming conventions](#) before submitting your package. The authoritative name of your package will be taken from the composer.json file inside the main branch of your repository, and it can not be changed after that.

Do not submit forks of existing packages. If you need to test changes to a package that you forked to patch, use [VCS Repositories](#) instead. If however it is a real long-term fork you intend on maintaining feel free to submit it.

If you need help or if you have any questions please get in touch with the Composer [community](#).

[About Packagist](#) [Statistics](#) [API](#) [Status](#)
[Atom/RSS Feeds](#) [Browse Packages](#) [Mirrors](#)

[✉](#) [🐦](#) [👤](#)

Packagist maintenance and hosting is provided by [Private Packagist](#)

packagist.org/packages/submit

Packagist *The PHP Package Repository* Browse Submit wild-garlic

Search packages...

Submit package

Repository URL (Git/Svn/Hg)

Notice: One or more similarly named packages have already been submitted to Packagist. If this is a fork read the notice above regarding VCS Repositories.

Similarly named packages:

- [1452479141/hello-world](#)
- [3alampro/hello-world](#)
- [44301/hello-world](#)
- [aayushi/hello-world](#)
- [abinmt/hello-world](#)

And 304 more

The package name found for your repository is: **wild-garlic/hello-world**, press Submit to confirm.

Submit

Trying to share private code?

Use [Private Packagist](#) to share code through Composer without publishing it for everyone on Packagist.org.

Please make sure you have read the package [naming conventions](#) before submitting your package. The authoritative name of your package will be taken from the composer.json file inside the main branch of your repository, and it can not be changed after that.

Do not submit forks of existing packages. If you need to test changes to a package that you forked to patch, use [VCS Repositories](#) instead. If however it is a real long-term fork you intend on maintaining feel free to submit it.




If you need help or if you have any questions please get in touch with the [Composer community](#).

About Packagist
Atom/RSS Feeds

Statistics
Browse Packages

API
Mirrors

Status

★ wild-garlic/hello-world

↓ composer require wild-garlic/hello-world

A basic "Hello, World" example.

- Abandon
- Delete
- Update
- Edit

Maintainers



Details

github.com/wild-garlic/hello-world

[Source](#)

[Issues](#)

- Installs: 0
- Dependents: 0
- Suggesters: 0
- Security: 0
- Stars: 0
- Watchers: 1
- Forks: 0
- Open Issues: 4

dev-main 2023-10-31 00:33 UTC

requires

- php: ^8.1

provides

None

requires (dev)

- ramsey/devtools: ^2.0

conflicts

None

suggests

None

replaces

None

dev-main ✕

dev-longhorn2023-minimal ✕

This package is auto-updated.

Last update: 2023-10-31 00:46:09 UTC [View Log](#)



ramsey@Celebrian:~/Developer/Repos/wild-garlic/hello-world

hello-world on  main via  v8.2.10

>

—

wild-garlic/hello-world Changelog

All notable changes to this project will be documented in this file.

The format is based on [[Keep a Changelog](https://keepachangelog.com/en/1.1.0/)](https://keepachangelog.com/en/1.1.0/) and this project adheres to [[Semantic Versioning](https://semver.org/spec/v2.0.0)](https://semver.org/spec/v2.0.0)

0.1.0 - 2023-10-31

Added

- Basic greeting functionality.

Changed

- Nothing.

Deprecated

NORMAL ↵ main ↻ <GELOG.md mar... utf-8[unix] 56W < 42% L_N:12/28≡ %:31

"CHANGELOG.md" 28L, 443B written



ramsey@Celebrian:~/Developer/Repos/wild-garlic/hello-world

hello-world on main [+] via v8.2.10 took 3s

> git commit -m "chore: prepare version 0.1.0"

pre-commit:

- composer validate : skipped
- composer normalize --dry-run : skipped
- composer dev:lint:syntax -- {\$STAGED_FILES|of-type:php} : skipped
- composer dev:lint:style -- {\$STAGED_FILES|of-type:php} : skipped

prepare-commit-msg:

- \Ramsey\CaptainHook\PrepareConventionalCommit : done

commit-msg:

- \Ramsey\CaptainHook\ValidateConventionalCommit : done

post-commit:

- hook is disabled

[main 21fae27] chore: prepare version 0.1.0

1 file changed, 2 insertions(+), 2 deletions(-)

hello-world on main [↑] via v8.2.10

>

_

★ wild-garlic/hello-world

↓ composer require wild-garlic/hello-world

A basic "Hello, World" example.

- Abandon
- Delete
- Update
- Edit

Maintainers



Details

github.com/wild-garlic/hello-world

Source

Issues

- Installs: 0
- Dependents: 0
- Suggesters: 0
- Security: 0
- Stars: 0
- Watchers: 1
- Forks: 0
- Open Issues: 4

0.1.0 2023-10-31 01:24 UTC

requires

- php: ^8.1

provides

None

requires (dev)

- ramsey/devtools: ^2.0

conflicts

None

suggests

None

replaces

None

- dev-main ✕
- 0.1.0 ✕
- dev-longhorn2023-minimal ✕

This package is auto-updated.

Last update: 2023-10-31 01:25:25 UTC [View Log](#)

PUBLISH OUR PACKAGE

Review: Publish to Packagist

- Create package at packagist.org/packages/submit.
- Edit the CHANGELOG.
- Commit the changes, tag the release, and push.

```
git add -p
git commit -m "chore: prepare version 0.1.0"
git tag -m "Version 0.1.0" 0.1.0
git push origin main 0.1.0
```

- Packagist automatically picks up the new version!



ramsey@Celebrian:~/Developer/Sandbox

~/Developer/Sandbox

> _

PUBLISH OUR PACKAGE

Review: Use our package in another project

- We can use our package in another project!

```
mkdir my-new-project
cd my-new-project
composer require wild-garlic/hello-world
```

- composer.json

```
{
    "require": {
        "wild-garlic/hello-world": "^0.1.0"
    }
}
```

MAINTENANCE

LET'S EXPLORE

What's included in the starter kit?

- [CaptainHook](#) and Git hooks
- [phly/keep-a-changelog](#)
- [madewithlove/license-checker](#)
- [Psalm](#) and [PHPStan](#)
- [PHPUnit](#)
- [PHP_CodeSniffer](#)
- [ramsey/devtools](#)
- [ramsey/conventional-commits](#)
- [ramsey/composer-repl](#)
- GitHub Actions workflow for continuous integration
- GitHub Actions workflow to auto-merge Dependabot pull requests

LET'S EXPLORE

What's included in the starter kit?

- Useful commands:

```
composer list
composer help dev:changelog
composer help dev:license
composer help dev:analyze:phpstan
composer help dev:analyze:psalm
composer help dev:lint:style
composer help dev:test:unit
composer test
composer repl
```

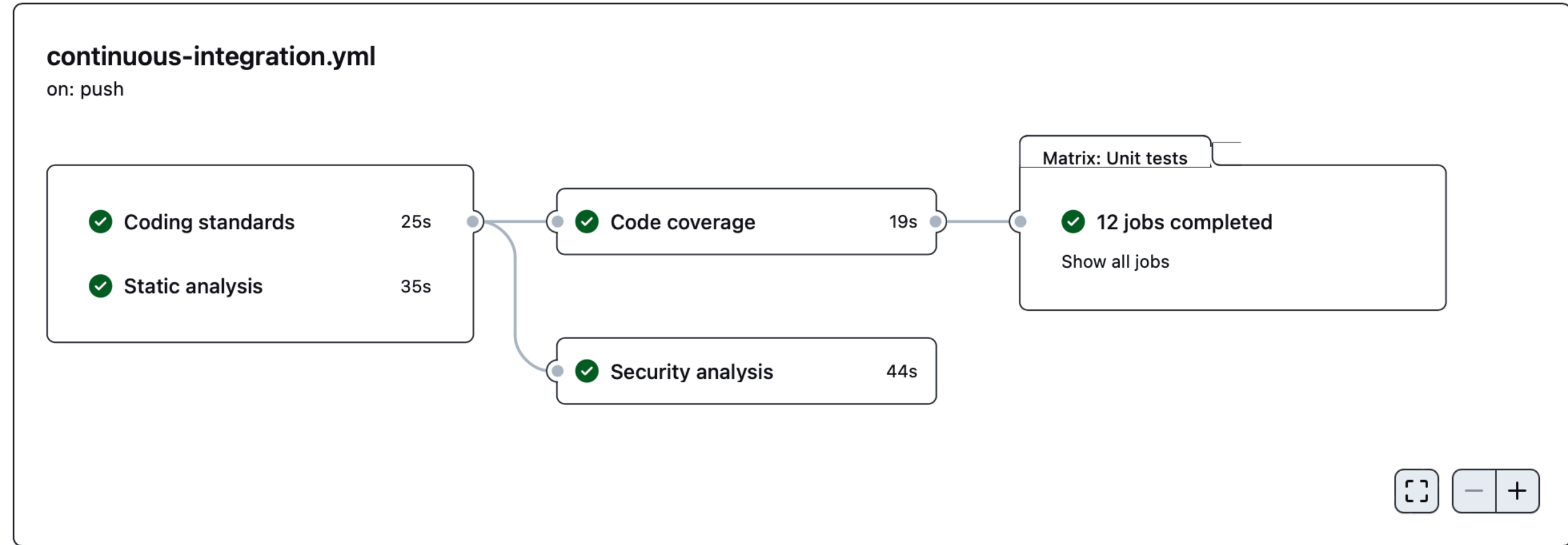
← Continuous Integration

🟢 chore: prepare version 0.1.0 #9 Re-run all jobs

🏠 Summary

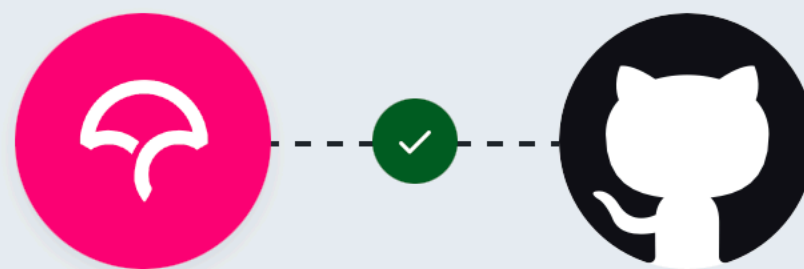
- Jobs
- 🟢 Coding standards
 - 🟢 Static analysis
 - 🟢 Security analysis
 - 🟢 Code coverage
 - 🟢 Unit tests (8.1, macos-latest, lo...)
 - 🟢 Unit tests (8.1, macos-latest, hi...)
 - 🟢 Unit tests (8.1, ubuntu-latest, lo...)
 - 🟢 Unit tests (8.1, ubuntu-latest, hi...)
 - 🟢 Unit tests (8.1, windows-latest, l...)
 - 🟢 Unit tests (8.1, windows-latest, ...)
 - 🟢 Unit tests (8.2, macos-latest, lo...)
 - 🟢 Unit tests (8.2, macos-latest, hi...)
 - 🟢 Unit tests (8.2, ubuntu-latest, lo...)
 - 🟢 Unit tests (8.2, ubuntu-latest, hi...)
 - 🟢 Unit tests (8.2, windows-latest, ...)
 - 🟢 Unit tests (8.2, windows-latest, ...)

Triggered via push yesterday	Status	Total duration	Artifacts
wild-garlic pushed -o 21fae27 0.1.0	Success	3m 36s	-



If you didn't do it earlier, go register at
Codecov using your GitHub account.


codecov.io/login/gh



Codecov by **Codecov** would like permission to:


 **Verify your GitHub identity (wild-garlic)**

 **Know which resources you can access**

 **Act on your behalf**

[? Learn more](#)

Resources on your account


 **Email addresses (read)**
View your email addresses

[Learn more about Codecov](#)

Cancel

Authorize Codecov

Authorizing will redirect to
<https://codecov.io>

 **Not owned or operated by GitHub**

 Created **8 years ago**

 **More than 1K GitHub users**

app.codecov.io/gh/wild-garlic

Docs Support Blog Feedback

Configure Codecov's GitHub app [↗](#)
Codecov will use the integration to post statuses and comments.

wild-garlic

Repos Analytics Members Plan Settings [Try Codecov Pro](#)

All Active Inactive Search [Can't find your repo? Try syncing...](#) Most recent commit


There are no repos detected

© 2023 Sentry [Terms](#) [Privacy](#) [Security](#) [GDPR](#) [Pricing](#) [Support](#) [Docs](#) [Feedback](#)

github.com/apps/codecov

Codecov GitHub App

Type to search



Codecov

GitHub App

Codecov provides highly integrated tools to group, merge, archive and compare coverage reports. Whether your team is comparing changes in a pull request or reviewing a single commit, Codecov will improve the code review workflow and quality.

Code coverage done right.®

1. Upload coverage reports from your CI builds.
2. Codecov merges all builds and languages into one beautiful coherent report.
3. Get commit statuses, pull request comments and coverage overlay via our browser extension.

When Codecov merges your uploads it keeps track of the CI provider (inc. build details) and user specified context, e.g. `#unittest` ~ `#smoketest` or `#oldcode` ~ `#newcode`. You can track the `#unittest` coverage independently of other groups. [Learn more here](#)

Through **Codecov's Browser Extension** reports overlay directly in GitHub UI to assist in code review in [Chrome](#) or Firefox (<https://addons.mozilla.org/en-US/firefox/addon/codecov/>)

Highly detailed pull request comments and customizable commit statuses will improve your team's workflow and code coverage incrementally.

Developer

[codecov](#)

[Website](#)


Codecov is provided by a third-party and is governed by separate terms of service, privacy policy, and support documentation.

[Report abuse](#)

github.com/apps/codecov/installations/new/permissions?target_id=149326471

Codecov Installing Codecov

Search Type to search



Install Codecov

Install on your personal account Wild Garlic

for these repositories:

- 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 administration, code, issues, and metadata
- ✓ **Read and write** access to checks, commit statuses, and pull requests



 **wild-garlic** ▼


Repos [Analytics](#) [Members](#) [Plan](#) [Settings](#)

 [Try Codecov Pro](#)

All [Active](#) [Inactive](#)

Can't find your repo? Try [re-syncing](#)

Most recent commit ▼

Name	Last updated	Tracked lines	Test coverage
 hello-world			Setup repo >



← Continuous Integration

chore: prepare version 0.1.0 #9

Re-run all jobs

Summary

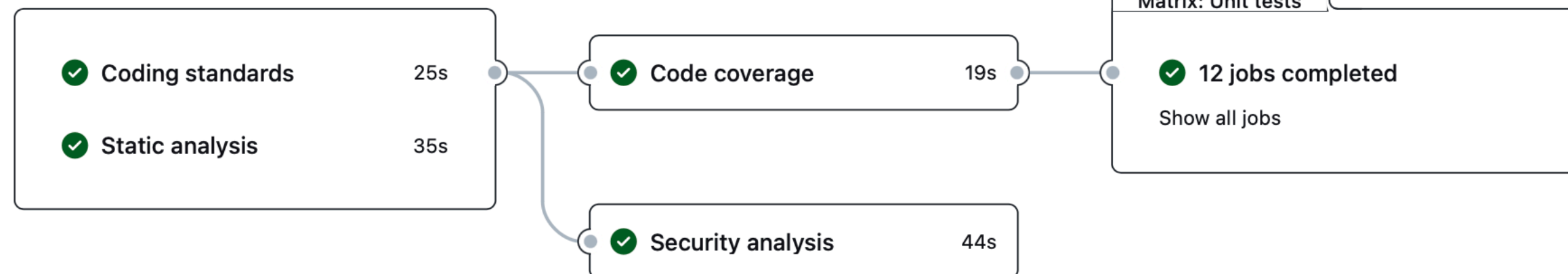
Jobs

- ✓ Coding standards
- ✓ Static analysis
- ✓ Security analysis
- ✓ Code coverage
- ✓ Unit tests (8.1, macos-latest, lo...
- ✓ Unit tests (8.1, macos-latest, hi...
- ✓ Unit tests (8.1, ubuntu-latest, lo...
- ✓ Unit tests (8.1, ubuntu-latest, hi...
- ✓ Unit tests (8.1, windows-latest, l...
- ✓ Unit tests (8.1, windows-latest, ...
- ✓ Unit tests (8.2, macos-latest, lo...
- ✓ Unit tests (8.2, macos-latest, hi...
- ✓ Unit tests (8.2, ubuntu-latest, lo...
- ✓ Unit tests (8.2, ubuntu-latest, hi...
- ✓ Unit tests (8.2, windows-latest, ...
- ✓ Unit tests (8.2, windows-latest, ...

Triggered via push yesterday	Status	Total duration	Artifacts
wild-garlic pushed -o 21fae27 0.1.0	Success	3m 36s	-

continuous-integration.yml

on: push





 **wild-garlic** ▼


Repos [Analytics](#) [Members](#) [Plan](#) [Settings](#)

 [Try Codecov Pro](#)

All [Active](#) [Inactive](#)

Can't find your repo? Try [re-syncing](#)

Most recent commit ▼

Name	Last updated	Tracked lines	Test coverage
 hello-world	about 21 hours ago	2	100.00%



wild-garlic / hello-world / main

Coverage Flags Commits Pulls Settings

Branch Context
main

Coverage on branch
100.00%
2 of 2 lines covered

3 Months trend
+100.00%

Hide Chart



Code tree File list hello-world

Search for files

Files ↑	Tracked lines	Covered	Partial	Missed	Coverage %
src	2	2	0	0	100.00%

Badges?

We need those stinkin' badges!

✓ chore: display project badges

Browse files

main

wild-garlic committed 41 minutes ago Verified

1 parent [21fae27](#) commit [4b1a3ef](#)

Showing 1 changed file with 0 additions and 6 deletions.

Split Unified

```

6 README.md
@@ -4,11 +4,6 @@
4 4     <strong>A basic "Hello, World" example.</strong>
5 5     </p>
6 6
7 7     - <!--
8 8     - TODO: Make sure the following URLs are correct and working for your project.
9 9     -     Then, remove these comments to display the badges, giving users a quick
10 10    -     overview of your package.
11 11    -
12 7     <p align="center">
13 8     <a href="https://github.com/wild-garlic/hello-world"></a>
@@ -18,7 +13,6 @@ TODO: Make sure the following URLs are correct and working for your project.
18 13    <a href="https://codecov.io/gh/wild-garlic/hello-world"></a>
19 14    <a href="https://shepherd.dev/github/wild-garlic/hello-world"></a>
20 15    </p>
21 21    - -->
22 16
23 17
24 18    ## About

```

wild-garlic/hello-world [↗](#)

A basic "Hello, World" example.

source [wild-garlic/hello-world](#) release [v0.1.0](#) php [^8.1](#) license [MIT](#)  build [passing](#)  codecov [100%](#) type-coverage [100%](#)

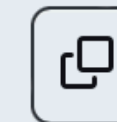
About [↗](#)

This project adheres to a [code of conduct](#). By participating in this project and its community, you are expected to uphold this code.

Installation [↗](#)

Install this package as a dependency using [Composer](#).

```
composer require wild-garlic/hello-world
```



Contributing [↗](#)

Contributions are welcome! To contribute, please familiarize yourself with [CONTRIBUTING.md](#).

Coordinated Disclosure [↗](#)

Keeping user information safe and secure is a top priority, and we welcome the contribution of external security researchers. If you believe you've found a security issue in software that is maintained in this repository, please read [SECURITY.md](#) for instructions on submitting a vulnerability report.

Copyright and License [↗](#)

wild-garlic/hello-world is copyright © [Ben Ramsey](#) and licensed for use under the terms of the MIT License (MIT). Please see [LICENSE](#) for more information.

BUILDING COMMUNITY

CONTRIBUTING.md

- Lets others know you accept contributions.
- Provides information on how to contribute to your project.
- Includes details such as setting up environments for testing.
- The starter kit provides an example you can use out-of-the-box.
- More info: mozillascience.github.io/working-open-workshop/contributing

BUILDING COMMUNITY

CODE_OF_CONDUCT.md

- A code of conduct indicates that you care about building a friendly and welcoming community.
- Sets the guidelines for what you expect from participants and what participants can expect from you.
- The starter kit includes a few you may use out-of-the-box.
- More info: opensource.guide/code-of-conduct

BUILDING COMMUNITY

SECURITY.md

- SECURITY.md is your project's vulnerability disclosure policy (VDP).
- It describes how you'll accept and respond to security vulnerability reports.
- The starter kit includes a policy created using [HackerOne's policy builder](#).
- You'll want to customize this to suit your, or your organization's, needs.

**YOU'RE A PHP PACKAGE
DEVELOPER!**

EVALUATING QUALITY

Things I look for in a package

- Are there tests? How comprehensive are they?
- What percentage of the code is covered by tests?
- Does the project use static analysis tools as part of its CI workflows?
- Does the project have CI workflows?
- Must have a README, CHANGELOG, and LICENSE.
- Are the maintainers friendly and responsive to issues and pull requests?

*Does your package
check all the boxes?*

THANK YOU!

Keep in touch



ben.ramsey.dev



phpc.social/@ramsey



github.com/ramsey



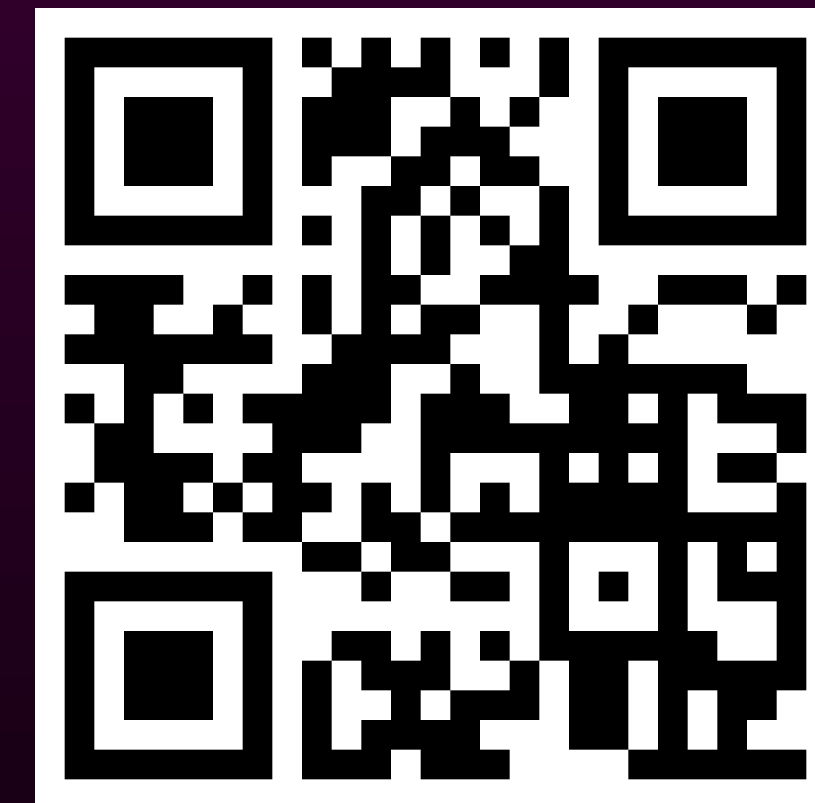
speakerdeck.com/ramsey



www.linkedin.com/in/benramsey



ben@ramsey.dev



joind.in/talk/bf3aa



ATTRIBUTION

- Fonts
 - Archivo Black by Omnibus-Type, SIL Open Font License, Version 1.1
 - DM Mono by Colophon Foundry, SIL Open Font License, Version 1.1
 - Playfair Display by Claus Eggers Sørensen, SIL Open Font License, Version 1.1
 - Saira by Omnibus-Type, SIL Open Font License, Version 1.1
- OpenMoji
 - “person shrugging” by Johanna Wellnitz, CC BY-SA 4.0
 - “winking face” by Emily Jäger, CC BY-SA 4.0