Git hooks and make targets

The local checks of the Exponential repository - the pre-commit, commit-msg and pre-push hooks in .githooks/, every make target from make check to make lint80-all, the file manifest share/filelist.md5, and the commit...

Source file: doc/eng-us/help/php-developers/git-hooks-and-make-targets.md Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/php-developers/git-hooks-and-make-targets.md

This page is for PHP developers who commit to the Exponential repository, or who want their extension's checks to look the same. The repository checks every change in the same way on your machine (git hooks and `make`) and in CI (the Quality workflow), so a check that passes locally passes in CI.

Contents

1. Set it up once
2. The hooks
3. The make targets
4. The file manifest
5. Commit messages
6. Skipping a hook

Set it up once

Shell

```
make devtools   # PHP_CodeSniffer, PHPCompatibility and PHPStan into .devtools/ (composer.dev.jso
n)
make hooks      # git config core.hooksPath .githooks
```

`make hooks` prints `Git hooks enabled: pre-commit, commit-msg, pre-push`. Without `make devtools` the hooks still run, and skip code style and PHPStan with a hint.

The hooks

Hook	Check	Refuses
pre-commit	`php -l` on every staged PHP file	a syntax error
pre-commit	`php -l` with PHP 8.0 when one is found (`EXP_PHP80`, `php8.0`, `php80`, `/opt/plesk/php/8.0/bin/php`)	syntax of PHP 8.1 and later
pre-commit	`node --check` on staged JavaScript (not `*.min.js`), when node is installed	a syntax error
pre-commit	`php bin/php/checkmanifest.php --staged`	a staged file whose checksum is missing or stale in `share/filelist.md5`
pre-commit	`php bin/php/checkcodestyle.php --staged`	a change that adds code style violations

Hook	Check	Refuses
commit-msg	the subject format	a subject without one of the five prefixes
pre-push	PHPStan against `phpstan-baseline.neon`	a finding that is not in the baseline

The make targets

Target	Does
`make check`	`lint`, `manifest-check`, `phpcs` and `phpstan`, as CI does
`make lint`	`php -l` on the PHP files changed since `origin/main` (`BASE=<ref>` for another base)
`make lint80`	the same with PHP 8.0 (`PHP80=/path/to/php` or `EXP_PHP80`)
`make lint80-all`	PHP 8.0 syntax of every PHP file in git, 8 in parallel, as CI's `php80` job (about 25 seconds)
`make manifest-check`	`php bin/php/checkmanifest.php --all`: every entry of `share/filelist.md5` exists and matches
`make manifest-fix`	`php bin/php/checkmanifest.php --fix`: rewrites stale checksums, adds missing files, drops removed ones
`make phpcs`	`php bin/php/checkcodestyle.php --base=$(BASE)`: code style of changed files, only new violations fail
`make phpstan`	PHPStan, only findings missing from the baseline fail
`make phpstan-baseline`	accepts all current findings as the new baseline (rarely, in a change of its own)

The file manifest

`share/filelist.md5` lists `md5 path` for the files of the distribution. The admin's upgrade check (**Setup > Upgrade check**, file consistency) compares the installed files against it, so a stale entry shows up there as a "modified" file on every installation. After changing or adding a file:

Shell

```
make manifest-fix
git add share/filelist.md5
```

`var/`, `extension/ezoe/` and the manifest itself are outside it. In CI the manifest check blocks on a push to `main` and is a warning on a pull request, where the person merging refreshes it (File consistency check).

Commit messages

The subject starts with what the change does:

Prefix	For
`Added:`	a new feature, file, class or capability
`Updated:`	changed or improved behavior

Prefix	For
`Fixed:`	a bug fix
`Removed:`	deleted features, files or dependencies
`Renamed:`	renamed files, classes or settings

Git's own `Merge`, `Revert`, `fixup!` and `squash!` subjects pass too. For commits by the project's own authors (addresses in `.githooks/commit-msg-authors`) a wrong subject fails CI; for a contributor it is a warning. Write the product name as Exponential in messages, and describe why and what changed (Commit messages).

Shell

```
sh .githooks/check-commit-range origin/main HEAD                                    # as CI checks a range
EXP_COMMIT_MSG_STRICT_ALL=1 sh .githooks/check-commit-range origin/main HEAD        # every commit as a project commit
```

Skipping a hook

`git commit --no-verify` and `git push --no-verify` skip the hooks in an emergency. CI runs the same checks, so the problem returns there; fix it rather than skip it.

Related pages

- Quality checks
- CI and reproducing a red check
- Contributing PHP changes
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

> **Become an Exponential Partner**
>
> Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.
>
> Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.