Setting up a PHP development checkout

A working checkout for PHP work on Exponential - clone, Composer, a SQLite installation, the dev environment name, the development tools in .devtools/, the git hooks, debug settings and the commands you run every day.

Source file: `doc/eng-us/help/php-developers/a-php-development-checkout.md` Last changed 2026-11-04
https://doc.exponential.earth/exphelp/viewer/help/php-developers/a-php-development-checkout.md

This page is for PHP developers who want a checkout of Exponential to write and test PHP code in: kernel fixes, extensions, tests. It goes a step further than the application creators' setup page (Setting up a development environment): the development tools, the git hooks and a PHP 8.0 binary for compatibility checks.

Contents

1. What you need
2. Clone and install
3. Name the environment
4. The development tools
5. The git hooks
6. A PHP 8.0 binary
7. Debug settings for development
8. Daily commands

What you need

Tool	Why
PHP 8.0 or newer on the command line	the kernel supports 8.0 to 8.5; PHPUnit 13 in the installed `vendor/` needs 8.4
Composer 2	the dependencies and the development tools
Git	the hooks and the change-based checks compare against `origin/main`
Node.js (optional)	`node --check` of changed JavaScript in the pre-commit hook
A second PHP, 8.0 (optional)	`make lint80`, the oldest supported syntax

Clone and install

Shell

```
git clone https://github.com/se7enxweb/exponential.git exp-dev
cd exp-dev
composer install
php bin/php/console exp:install --apply --package=sevenx_multisite_clean --url=http://127.0.0.1:8088
```

`composer install` runs `bin/php/ezpgenerateautoloads.php` afterwards (`"post-install-cmd"` in `composer.json`). The install uses SQLite unless you pass `--db=` (The console install). Reinstall with `--force` when you need a clean state.

Name the environment

Copy `config.env.php.example` to `config.env.php` and set the environment name of this machine:

PHP

```
<?php
define( 'EXP_ENV', 'dev' );
```

With `dev` (or `test`), `index.php` shows PHP errors in the browser, and `eZINI` also reads a variant of every settings file with the name after `.ini`, for example `settings/override/site.ini.dev.append.php`, after the standard file of the same layer. Each environment has its own INI cache. `config.env.php` is listed in `.gitignore`; never commit it.

The development tools

Shell

```
make devtools   # PHP_CodeSniffer, PHPCompatibility and PHPStan into .devtools/ (composer.dev.jso
n), never into vendor/
```

The tools are resolved for PHP 8.0.30 from `composer.dev.json`, so they run on every supported PHP. After this, `.devtools/vendor/bin/phpcs` and `.devtools/vendor/bin/phpstan` exist and the make targets use them (Git hooks and make targets).

The git hooks

Shell

```
make hooks      # git config core.hooksPath .githooks
```

Hook	Checks
pre-commit	`php -l` of staged PHP files, PHP 8.0 syntax when a PHP 8.0 binary is found, `node --check` of staged JavaScript, the file manifest, the code style of staged PHP files
commit-msg	the subject starts with `Added:` , `Updated:` , `Fixed:` , `Removed:` or `Renamed:`
pre-push	PHPStan against the baseline

Without `make devtools` the hooks still check syntax, the manifest and the message, and skip code style and PHPStan with a hint.

A PHP 8.0 binary

PHP 8.0 is the oldest supported version (the stock PHP of RHEL 9); PHP 7.4 is usable without support (`make lint74` checks it). The pre-commit hook and `make lint80` find PHP 8.0 as `EXP_PHP80`, `php8.0` or `php80` on the `PATH`, or `/opt/plesk/php/8.0/bin/php`:

Shell

```
make lint80                                  # PHP 8.0 syntax of the files changed since origin/
main
make lint80-all PHP80=/path/to/php8.0        # every PHP file in git, as CI runs it
```

Without one, CI's `php80` job is your only PHP 8.0 check (Writing code that runs on PHP 8.0).

Debug settings for development

In `settings/override/site.ini.append.php` of the development machine only:

INI

```
[DebugSettings]
DebugOutput=enabled

[TemplateSettings]
DevelopmentMode=enabled
ShowTemplatePathComments=enabled
```

The debug bar then shows queries, templates, timings and settings for each page (The debug bar for PHP developers). Put these lines in a `site.ini.dev.append.php` instead, and they can never reach a machine whose environment is not `dev`.

Daily commands

Command	When
`php -l <file.php>`	after every edit
`php bin/php/ezpgenerateautoloads.php -e`	after adding, renaming or moving an extension class
`php bin/php/ezcache.php --clear-tag=ini`	after a settings change
`php bin/php/ezcache.php --clear-all`	after a class or template path change, when output looks stale
`vendor/bin/phpunit <test file>`	the tests of what you changed
`make check`	syntax, manifest, code style and PHPStan before you push
`php bin/php/console exp:velocity deploy`	under Velocity: autoloads, caches, workers, in the right order

Add `--allow-root-user` to the cache and console commands only when you work as root.

Related pages

- Develop with Git
- Setting up a development installation
- Environment settings
- 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.