Development environment

For CMS developers, a working development setup in a few commands. It covers the code as Git clones, a one-command install on SQLite, Velocity with PHP's built-in server engine, debug output and template settings, a…

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

This page is for developers setting up the machine they build sites on. A development installation takes a few commands: the kernel and extensions as Git clones, a database in one SQLite file, and a web server started by one command. The book's chapter Setting up a development installation has every detail; this page is the CMS developer's summary with the project around it.

Contents

1. What you need
2. Step 1: the code as Git clones
3. Step 2: install with one command
4. Step 3: serve it
5. Step 4: make mistakes visible
6. Step 5: your project's extensions
7. Working as a team
8. Rules that keep development and live apart
9. When something goes wrong

What you need

PHP 8.0 or later (PHP 7.4 usable without support) with the extensions the root `composer.json` requires (dom, mbstring, json, iconv, simplexml and others; SQLite 3 for the default database, GD or ImageMagick for images), Composer and Git (Tools you need, Requirements).

Step 1: the code as Git clones

Shell

```
composer config --global 'preferred-install.se7enxweb/*' source
git clone https://github.com/se7enxweb/exponential.git mysite
cd mysite
git switch -c my-dev
composer install
```

With `preferred-install` set to `source`, every `se7enxweb/*` extension is a Git clone under `extension/<name>` with its history, so you can read why code is the way it is and send fixes back (Develop with Git, Getting the code).

Step 2: install with one command

Shell

```
php bin/php/console exp:install --apply
```

Without options this installs the `sevenx_multisite` site package on SQLite in `eng-US` and prints the site address, the administration login and a generated administrator password (also in `var/log/initial-admin-password`, readable by its owner only). Other databases and packages: The console install. It refuses to run over an existing installation unless `--force` is given; never point it at a database that matters.

A shortcut many developers add: `ln -s bin/php/console console`, then `./console list`.

Step 3: serve it

Shell

```
./console exp:velocity start
```

Velocity, the Exponential application server, ships with `settings/velocity.ini [ServerSettings] Engine=php`: it runs PHP's built-in server (`php -S`) with the router `bin/php/velocity-router.php`, on the port of `[PHPServerSettings]` (8087 shipped). Open the address the installer printed. The persistent `qbix` engine, FrankenPHP and Apache or nginx with PHP-FPM are the options for production and for testing like production (Serving the site, Lesson 13.2: Velocity for developers).

Step 4: make mistakes visible

Shell

```
./console exp:ini set site.ini/DebugSettings/DebugOutput enabled global --allow-root-user
./console exp:ini set site.ini/TemplateSettings/ShowUsedTemplates enabled global --allow-root-use
r
./console exp:ini set site.ini/TemplateSettings/TemplateCompile disabled global --allow-root-user
php bin/php/ezcache.php --clear-tag=ini --allow-root-user
```

Errors, SQL and the templates used appear below each page, and template edits show at once. On any machine others can reach, limit debug output to your address (`DebugByIP`, `DebugIPList[]`) (Debug output and logs for developers, Debug bar). Logs are in `var/log/`.

Step 5: your project's extensions

Shell

```
mkdir -p extension/exp_mysite/settings extension/exp_mysite/design/exp_mysite/templates
git -C extension/exp_mysite init
./console exp:ini add 'site.ini/ExtensionSettings/ActiveExtensions[]' exp_mysite global --allow-r
oot-user
```

What goes in them: Extensions you will write. Classes made in the administration belong in packages in that repository (Creating a content class).

Working as a team

- One development installation per developer, each from the same project repositories.
- Classes and demo content travel in packages; layouts in `exp:layouts` transfer files; settings and templates in the extension. Rebuild a developer machine from these, never from a copy of live.
- A shared staging installation, deployed like live (Deploying a site), is where editors and testers see work before it goes live.
- Run the project's checks before sharing a change (Checks before you share a change).

Rules that keep development and live apart

Rule	Why
Never develop on the live site	a typo in a template or a settings file takes the page down
Never copy `settings/override/` between machines	it holds passwords and host-specific values
Use copies of live content only when you must	live content can hold personal data
Test accounts and mail use `example.invalid`	no test may mail a real person
Turn debug output off before going live	it reveals paths, SQL and settings

When something goes wrong

Symptom	Check
`composer install` stops with a PHP version error	`php -v`; Exponential 6 needs PHP 8.0 or later (PHP 7.4 usable without support)
A blank page	`var/log/error.log` and the PHP error log
A template change has no effect	`TemplateCompile`, the template and override caches
A new class is not found	`php bin/php/ezpgenerateautoloads.php -e`, then clear the caches and restart the server

More: When something goes wrong.

Related pages

- Your first hour as a developer
- Module 1: The development installation
- Demo mode for developers
- 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.