Software engineer FAQ

Short answers to the questions software engineers ask first about Exponential - language and versions, frameworks and PSRs, ORM and migrations, databases, servers, statics, caches, tests, releases, security reports...

Source file: `doc/eng-us/help/software-engineers/software-engineer-faq.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/software-engineers/software-engineer-faq.md

Short answers, each with the page that has the details.

Contents

1. Platform
2. Data
3. Runtime
4. Quality and releases
5. People and licenses

Platform

Which PHP versions? PHP 8.0 to 8.5. CI checks every file with PHP 8.0 and boots the kernel on it; PHPUnit runs on 8.1 to 8.5 (CI gates).

Is it built on Symfony or Laravel? No. The kernel is its own framework. The Symfony platform products embed it through the legacy bridge (Symfony and the legacy bridge).

Does it follow PSR-4 and PSR-12? No, deliberately: generated class maps instead of PSR-4, and its own code style checked by `phpcs.xml.dist` (Dependency management with Composer).

Why no type declarations in kernel code? The code must run on PHP 8.0 and keep its signatures, which extensions override (Static analysis and code style).

Where does my code go? In an extension, almost always (Kernel, lib and extensions: the layering).

Data

Is there an ORM? An active-record mapper, `eZPersistentObject`, one class per table; no unit of work, no lazy relations (Persistence and the database layer).

Are there migrations? Schema files (`.dba`) and SQL update files per driver, applied by hand; no runner records what ran (Schema changes and migrations).

Which database should we use? SQLite 3 for one machine, PostgreSQL for several nodes or the best-tested server; the official order is SQLite 3, PostgreSQL, MongoDB, MariaDB, Oracle, MySQL (Databases: choosing and running one).

Are prepared statements used? Statements are built as strings; every string is escaped with `escapeString()` and every number cast (Writing portable SQL).

Runtime

Which server should serve it? The project recommends Velocity; FrankenPHP is the production-ready alternative, and PHP-FPM behind any web server keeps working (Application servers).

Are statics safe under Velocity? Static properties are restored after each request, but warm-up state, globals kept on purpose, resources and accumulated registrations are your job (Concurrency, statics and long-lived processes).

Why did my change not show up? Usually the order: code must be loaded (PHP-FPM, Velocity restart) before the page caches are cleared (Deploying a change).

Is there a health endpoint? No single one; check the front page and the administration login from outside, and the status commands on the server (Observability).

Can it run on several web nodes? Yes, with a database server, the DFS cluster file handler for files and a shared session handler (Scaling and clustering).

Quality and releases

How do I run the tests? `php vendor/bin/phpunit` runs every suite without a database (DB-less tests and test suites).

Why is CI red on my pull request when my branch was green? It tests the merge with `main`; merge `main` into your branch (CI gates).

What comes after 0.0.4.9? `0.0.4.10` (Version numbers and immutable tags).

Can a broken tag be fixed? No. Publish the next version and say in its notes what was wrong.

Where do I find what changed? `doc/changelogs/` and the upgrade notes in `doc/eng-us/bc/6.0/` (Changelogs and upgrade notes).

People and licenses

Who maintains it? 7x leads the development; the Exponential Foundation operates and funds the project (Who is 7x?, Who is the Exponential Foundation?).

What license? The GNU GPL, version 2 or later, for the code; the GNU FDL 1.3 for the documentation.

What does contributing to the kernel require? A signed CLA, which assigns the copyright in the contribution to the Exponential Foundation and 7x; your own extensions are not affected (Contributing upstream).

How do I report a vulnerability? Privately, to the address in `SECURITY.md` (Security engineering overview).

Related pages

- Glossary for software engineers
- Application creator FAQ
- Software engineers: start here

- 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.