Code review checklist

A checklist for reviewing a change to the Exponential kernel or an extension - scope and history, layering and naming, compatibility with PHP 8.0 and every database and server, tests, caches and deploy, settings...

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

This page is a checklist for reviewing a change to Exponential or one of its extensions. The project does many review rounds before it accepts a change (Contributing upstream); a shared list makes each round shorter. Pair it with the Security review checklist.

Contents

1. Scope and history
2. Architecture
3. Compatibility
4. Runtime
5. Tests and CI
6. Settings, templates and styling
7. Documentation and release record

Scope and history

☐ One change per pull request: a fix and a refactoring are two.

☐ Commit subjects start with `Added:` , `Updated:` , `Fixed:` , `Removed:` or `Renamed:` ; the body says why and what (Git workflow).

☐ The product is named Exponential; kernel identifiers such as `eZContentObject` stay as they are.

☐ No tool attribution, private paths, credentials or machine-specific settings in commits.

☐ The branch is squashed into logical commits and rebased onto `main` while open.

Architecture

☐ The code is in the right layer (Kernel, lib and extensions: the layering): no new `lib/` dependency on `kernel/`, no HTML printed by a domain class.

☐ New code uses the `exp` prefix; nothing existing is renamed.

☐ An extension point is used before a kernel override (Events, extension points and plugins).

☐ Shared logic is a service class, not copied between a view, a command and a cronjob part.

☐ Settings are read through `eZINI`; debug output through `eZDebug`; text through `ezpI18n::tr()`.

☐ New classes have their class maps regenerated.

Compatibility

☐ Runs on PHP 8.0: no newer syntax or functions without a guard; no PHP 7+ type declarations in kernel code (Static analysis and code style).

☐ SQL works on every supported database; schema changes ship for every driver (Schema changes and migrations).

☐ No public signature, setting, template variable, command option or exit code changed without upgrade notes (Backward compatibility policy).

☐ New behavior is opt-in, or a safe default that keeps the old result.

Runtime

☐ Correct in a long-lived Velocity worker: no per-request state in statics, idempotent registrations, no `exit` (Concurrency, statics and long-lived processes).

☐ A new class named in a shipped INI file is guarded with `class_exists()`.

☐ Cache files are written through `eZClusterFileHandler`; caches have ids and tags and are invalidated (Cache invalidation and clearing).

☐ Commands keep `--help` and exit codes stable and offer `--dry-run` when they change data.

☐ Performance: no query in a loop where one query does; measure when in doubt (Performance measurement and benchmarks).

Tests and CI

☐ A test fails without the change and passes with it.

☐ Tests need no database unless they test SQL, and read nothing from a developer's installation.

☐ The tests are in a suite of `phpunit.xml`, or they never run.

☐ Quality and PHPUnit are green on the merge with `main` (CI gates).

☐ `share/filelist.md5` has the checksum of every added, changed or removed file (`make manifest-fix`).

Settings, templates and styling

☐ A new setting has its default and a comment in `settings/<file>.ini`; switches use `enabled/disabled`.

☐ Templates escape output and contain no SQL.

☐ Administration templates follow Styling the administration: no `<style>` or `style=""`, shared components and tokens, a dark value for every new token, module CSS in `stylesheets/modules/`.

☐ Changed pages checked in light and dark mode and at the widths the administration

supports.

Documentation and release record

- ☐ The feature page in `doc/eng-us/features/6.0/` is written or updated.
- ☐ The changelog lists the change (Changelogs and upgrade notes).
- ☐ A change that needs action from installations has a page in `doc/eng-us/bc/6.0/`.
- ☐ Help pages touched by the change pass `exp:help check` (Documentation as code).

Related pages

- Design review checklist
- Pull request etiquette
- Coding standards
- 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.