Checklisten für PHP-Entwickler

Kopierbare Checklisten für PHP-Arbeit an Exponential - bevor Sie schreiben, für jede PHP-Datei, für eine neue Klasse, eine View, einen Befehl oder Cronjob-Teil, einen Datentyp oder ein Workflow-Ereignis, Sicherheit...

Source file: `doc/ger-de/hilfe/php-entwickler/checklisten-fuer-php-entwickler.md` Last changed 2026-11-04
https://doc.exponential.earth/exphelp/viewer/help/php-developers/php-developer-checklists.md

Diese Seite richtet sich an PHP-Entwickler, die ihre Arbeit in einem Durchgang prüfen möchten. Jede Liste verweist auf die Seite, die sie erklärt. Kopieren Sie eine Liste in die Beschreibung Ihres Pull Requests oder in Ihre Review-Notizen.

Inhalt

1. Bevor Sie schreiben
2. Jede PHP-Datei
3. Eine neue Klasse
4. Eine Modul-View
5. Ein Befehl oder Cronjob-Teil
6. Ein Datentyp oder Workflow-Ereignistyp
7. Sicherheit
8. Velocity und lange Läufe
9. Tests
10. Commit und Pull Request
11. Deployment

Bevor Sie schreiben

☐ Gibt es eine Einstellung, eine mitgelieferte Erweiterung oder einen Erweiterungspunkt, der es bereits tut? (Erweiterungspunkte aus PHP)

☐ Kernel-Änderung oder Erweiterung? (PHP-Änderungen beitragen)

☐ Das Präfix der neuen Namen ist gewählt (Klassenbenennung und das Präfix exp)

Jede PHP-Datei

☐ Beginnt mit `<?php`, Unix-Zeilenenden, Leerzeichen, keine Leerzeichen am Zeilenende

☐ `php -l` besteht unter dem neuesten PHP und unter PHP 8.0 (`make lint80`)

☐ Keine ungeschützte Syntax oder Funktion, die neuer als PHP 8.0 ist (Code schreiben, der unter PHP 8.0 läuft)

☐ `array( ... )`, Leerzeichen innerhalb von Klammern, geschweifte Klammern von Funktionen in eigener Zeile (Der Coding-Standard)

☐ Keine skalaren, Rückgabe- oder Eigenschafts-Typdeklarationen; Docblocks mit `@param` und `@return`

☐ `make phpcs` fügt keine Verstöße hinzu; `make phpstan` fügt keine Befunde hinzu

Eine neue Klasse

☐ Eine Klasse pro Datei, Dateiname ist der Klassenname in Kleinbuchstaben

☐ Eigenschaften deklariert (keine dynamischen Eigenschaften)

☐ `php bin/php/ezpgenerateautoloads.php -e` ausgeführt (Autoloads und Klassen-Maps)

☐ Keine statische Klasseneigenschaft außerhalb des Klassenrumpfs zugewiesen

☐ Eine Einstellung, die die Klasse nennt, geht erst live, nachdem die Klasse bereitgestellt ist

Eine Modul-View

☐ `module.php` nennt die `functions` der View (Richtlinie) (Schreiben Sie Ihre erste Modul-View)

☐ Als Klasse geschrieben, die `Exponential\Runnable\ModuleView` erweitert, die View-Datei eine Zeile

☐ `$Params` gecastet und validiert

☐ Datenänderungen nur per POST, über Modulaktionen

☐ HTML in einem Template, jeder Wert durch `|wash`

☐ POST, Weiterleitung, GET nach einem erfolgreichen Speichern; Weiterleitungsziele aus der Anfrage geprüft

Ein Befehl oder Cronjob-Teil

☐ Eine Klasse, die `Exponential\Runnable\Command` oder `CronjobPart` erweitert (Einen Konsolenbefehl schreiben, Einen Cronjob-Teil schreiben)

☐ Tag `@description` in der Einstiegsdatei

☐ Exit-Code `0` bei Erfolg, ein anderer bei einem Fehlschlag; still mit `-q`

☐ `--dry-run` für alles, was Daten ändert

☐ Begrenzte Arbeit pro Lauf, kurze Transaktionen, In-Memory-Caches zwischen Stapeln geleert

☐ Die Arbeit in einer Dienstklasse, die View, Befehl und Cronjob-Teil gemeinsam nutzen

Ein Datentyp oder Workflow-Ereignistyp

☐ Methodensignaturen exakt wie in `eZDataType` / `eZWorkflowEventType` (Einen Datentyp schreiben)

☐ Registriert mit `eZDataType::register()` oder `eZWorkflowEventType::registerEventType()` und in der INI-Datei

☐ Hin- und Rückweg von `toString()`/`fromString()` getestet

☐ `execute()` schnell, gibt einen Status zurück, wirft nie (Einen Workflow-Ereignistyp schreiben)

Sicherheit

☐ Eingaben über `eZHTTPTool`, `$Params` oder Modulaktionen gelesen; Zahlen gecastet, Auswahlwerte in Allow-Listen (Eingaben validieren und casten)

☐ SQL: Strings durch `escapeString()`, Zahlen gecastet, oder Bedingungen persistenter Objekte (SQL mit eZDB, sicher)

☐ Ausgaben für ihren Kontext maskiert; `htmlspecialchars()` mit ausdrücklichen Flags (Ausgaben aus PHP maskieren)

☐ Berechtigungen in Views, Fetch-Funktionen, Serverfunktionen, REST-Aktionen und Befehlen geprüft (Sicherheit in PHP-Code)

☐ POSTs tragen das Formular-Token (CSRF-Schutz und ezformtoken)

☐ Keine Geheimnisse in Code, URLs oder Logs

☐ Keine Inline-Skripte und kein `style=""` in erzeugtem Markup

Velocity und lange Läufe

☐ Keine statischen Variablen in Funktionen, keine `true`-Flags für einmaliges Registrieren (Fallen durch statischen Zustand)

☐ Caches nach Siteaccess, Sprache und Benutzer geschlüsselt, wo sie davon abhängen

☐ `eZExecution::cleanExit()` nicht innerhalb von `catch ( Exception )`

☐ Pfade aus `EXP_ROOT_DIR` oder `eZSys`, nicht aus `__DIR__` einer Kernel-Datei

☐ Zweimal hintereinander auf demselben Worker getestet, als zwei Benutzer

Tests

☐ Ein Test ohne Datenbank für die Logik (PHPUnit ohne Datenbank)

☐ Datenbanktests mit `@group database` markiert (Datenbank- und Live-Tests)

☐ In einem Test gesetzte INI-Werte werden in `tearDown()` wiederhergestellt

☐ Funktioniert mit PHPUnit 10.5 bis 13

☐ Eintrag in `tests/test-map.json`, wenn der Test die Quelldatei nicht wörtlich nennt

Commit und Pull Request

☐ Betreff beginnt mit `Added:` , `Updated:` , `Fixed:` , `Removed:` oder `Renamed:` ; der Text sagt, warum

☐ Das Produkt wird in der Nachricht Exponential geschrieben

☐ `share/filelist.md5` aktualisiert (`make manifest-fix`) und committet

☐ `make check` besteht

☐ CLA hinterlegt; Pull Request klein und fokussiert; gesquasht und auf `main` rebased (

PHP-Änderungen beitragen)

Deployment

- ☐ Autoloads neu generiert, INI-Cache geleert (`--clear-tag=ini`)
- ☐ PHP-FPM neu geladen, Velocity-Worker neu gestartet (`exp:velocity deploy`)
- ☐ Caches mit gerenderter Ausgabe nach dem Neuladen geleert
- ☐ Caches vorgewärmt (OPcache und Cache-Vorladen)
- ☐ Startseite und die geänderten Seiten geprüft; Logs gelesen

Verwandte Seiten

- Die Checkliste für das Upgrade
- Die Checkliste für das Release
- FAQ für PHP-Entwickler
- Freie Dokumentation: diese Seite darf unter der GNU FDL 1.3 frei kopiert, übersetzt und verbessert werden.

> **Werden Sie Exponential-Partner**
>
> Partner teilen ihre Informationen, stellen Mittel bereit, geben Beiträge zurück und entwickeln die Produkte mit uns, und so wächst Exponential. Eine Partner-Website und ein Dashboard sind für Mitte 2027 geplant: Werden Sie noch heute Partner, damit wir schneller wachsen und neue Funktionen früher liefern.
>
> Partner: hier beginnen | Wie Sie sich bewerben

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.