# Zehn Stolperfallen für Framework-Entwickler

Die zehn Fehler, die Entwickler aus Symfony, Laravel, Django, Rails und ähnlichen Frameworks in ihrem ersten Exponential-Projekt am häufigsten machen, jeweils mit Symptom, Ursache und Lösung - von nicht maskierter…

Source file: `doc/ger-de/hilfe/web-framework-entwickler/stolperfallen-fuer-framework-entwickler.md`
Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/web-framework-developers/pitfalls-for-framework-developers.md

Diese Seite richtet sich an Entwickler in ihren ersten Wochen mit Exponential. Jede Stolperfalle stammt aus einer Gewohnheit, die in einem anderen Framework richtig und hier falsch ist.

## Inhalt

1. Ausgabe ohne wash
2. Eine neue Klasse wird „nicht gefunden“
3. Eine Einstellung wirkt nicht
4. Eine Template-Änderung wird nicht angezeigt
5. Zustand sickert zwischen Requests durch
6. SQL in Inhaltstabellen schreiben
7. Eine Tabelle für etwas, das Inhalt sein sollte
8. Berechtigungen nur in Templates prüfen
9. Den Siteaccess auf der Kommandozeile vergessen
10. Vergleiche mit Symbolen

## 1. Ausgabe ohne wash

**Symptom:** Text mit `<` oder `&` zerstört die Seite, oder ein Skript wird ausgeführt. **Ursache:** Templates maskieren nicht automatisch. **Lösung:** `{$value|wash}` auf alles, was von Benutzern oder aus der Datenbank kommt (Die View-Schicht).

## 2. Eine neue Klasse wird „nicht gefunden“

**Ursache:** Klassen werden über erzeugte Class Maps gefunden, für Erweiterungsklassen nicht über PSR-4. **Lösung:** `php bin/php/ezpgenerateautoloads.php -e`, dann deployen; `--kernel` für Kernel-Klassen (Autoloads und Klassenbenennung).

## 3. Eine Einstellung wirkt nicht

**Ursachen:** der INI-Cache; oder der Wert wird von einer späteren Ebene überschrieben, zum Beispiel gewinnen die globalen Einstellungen einer Erweiterung gegenüber `settings/siteaccess/`. **Lösung:** `./console exp:ini where <file>/<block>/<key>`

`<siteaccess>`, dann `php bin/php/ezcache.php --clear-tag=ini` (Konfiguration).

## 4. Eine Template-Änderung wird nicht angezeigt

**Ursachen:** der View-Cache, ein Cache-Block oder der Override-Cache bei einer neuen Template-Datei. **Lösung:** Leeren Sie `template-override` nach dem Hinzufügen einer Template-Datei, leeren Sie `content` und `template-block`, wenn zwischengespeicherte Ausgaben veraltet sind; `exp:velocity deploy` erledigt all das in der richtigen Reihenfolge ( Caching).

## 5. Zustand sickert zwischen Requests durch

**Symptom:** Ein Benutzer sieht die Daten eines anderen Benutzers, oder ein Listener läuft zweimal, nur unter Velocity. **Ursache:** Persistente Worker behalten statische Eigenschaften und angehängte Listener. **Lösung:** kein Request-bezogener Zustand in statischen Eigenschaften; registrieren Sie Listener in den Einstellungen (Server und Deployment).

## 6. SQL in Inhaltstabellen schreiben

**Symptom:** Eine Änderung erscheint nicht, oder sie erscheint ohne Versionsgeschichte, Suche oder Cache-Aktualisierung. **Ursache:** Die Inhaltstabellen sind nicht Ihre Tabellen. **Lösung:** `eZContentFunctions` oder die Operationen, die Workflows ausführen und Caches leeren (Das Inhaltsmodell).

## 7. Eine Tabelle für etwas, das Inhalt sein sollte

**Symptom:** Sie bauen eine Administrationsseite, eine Revisionstabelle und eine Übersetzungstabelle. **Lösung:** Machen Sie daraus eine Inhaltsklasse und erhalten Sie alle drei geschenkt (ORM).

## 8. Berechtigungen nur in Templates prüfen

**Ursache:** Einen Link auszublenden ist keine Zugriffskontrolle. **Lösung:** Nennen Sie die Funktion in `functions` der View, prüfen Sie den Objektzugriff in der View, filtern Sie Listen in der Abfrage (Autorisierung).

## 9. Den Siteaccess auf der Kommandozeile vergessen

**Symptom:** Ein Befehl oder Cronjob-Teil findet Ihr Modul, Ihre Einstellung oder Ihren Jobtyp nicht. **Ursache:** Ihre Erweiterung ist nur in einem Siteaccess aktiv. **Lösung:** `-s <siteaccess>` (Kommandozeile).

## 10. Vergleiche mit Symbolen

**Symptom:** Ein `{if}` verhält sich willkürlich, und das Log zeigt einen Parser-Fehler. **Ursache:** Die Template-Sprache hat keine Infix-Operatoren. **Lösung:** `gt`, `lt`, `eq`, `ne`, `and`, `or` (Kein Infix-Vergleich).

## Verwandte Seiten

- Checklisten für Framework-Entwickler
- Häufige Fehler in Templates
- Wenn etwas schiefgeht
- 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.