

Symfony vs. Laravel

Как писать код, чтобы не было больно

Обо мне

- Backend разработчик в 4xxi
- Занимаюсь разработкой 7 лет
- Работал с Symfony 2+, Laravel 5, Yii 1
- Умею работать с легаси уровня РНР без ООП
- Умею в python, Go и в разные базы данных

Обо мне

- Backend разработчик в 4xxi
 - Занимаюсь разработкой 7 лет
 - Работал с Symfony 2+, Laravel 5, Yii 1
 - Умею работать с легаси уровня РНР без ООП
 - Умею в python, Go и в разные базы данных
-
- Не люблю Laravel

О чем сегодня поговорим

1. О недостатках Laravel, которых нет в Symfony
2. О Laravel комьюнити
3. О том, что фасады — это синглтоны
4. О магии и Хогвартсе
5. О жизни

Комьюнити

- Низкий уровень входа — низкая квалификация разработчиков
- Некоторые разработчики даже не знают стандартную библиотеку php, но уже собеседуются на миддлов
- Миллениалы изобретают паттерны



Фасад Laravel — почти синглтон



Что такое фасад

Фасад — это структурный паттерн проектирования, который предоставляет простой интерфейс к сложной системе классов, библиотеке или фреймворку.

Фасады в Laravel являются
статическими классами

Проблемы фасадов в Laravel

- Нет возможности нормальной подмены реализации для тестов
- Доступны в любом месте кода
- Фасады везде в документации, хотя разработчики говорят, что они представляют опасность
- Фасады вызывают привыкание



Проблемный фасад

Глобальная область видимости

- Фасады засоряют глобальную область видимости
- Любой сервис можно достать где угодно, через `app()->make()` или `resolve()`
- Привязывают бизнес-логику к фреймворку
- Не позволяют копировать код в другие проекты



Принцип SOLID (моя версия)

Если на собеседовании просят рассказать значение каждой буквы SOLID, в коде проекта, вероятно, будет насрано

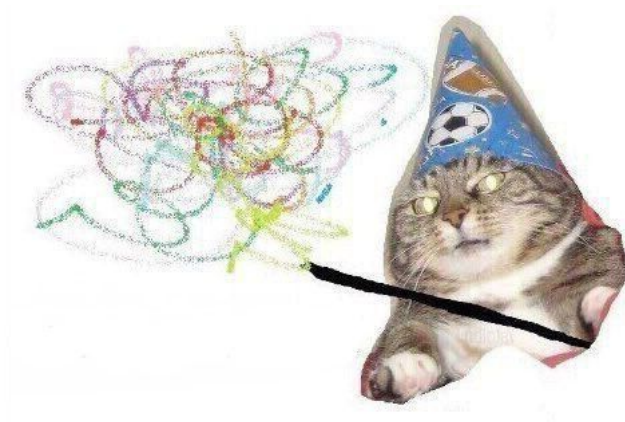
Eloquent — ActiveRecord модель

- Нарушает SRP
- Может ломать абстракции
- Вынуждает писать слишком много кода для изоляции логики работы с бд



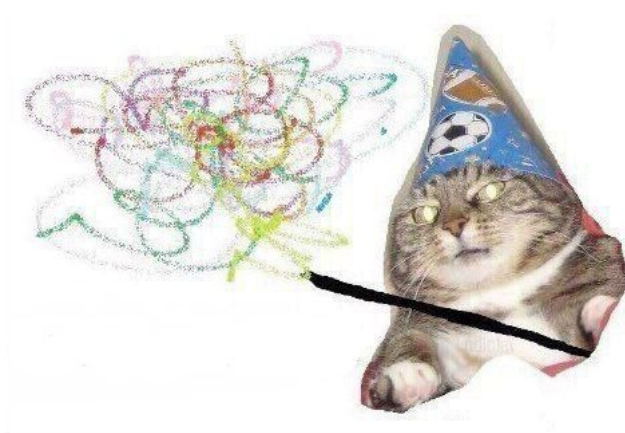
Магия

- Усложняет понимание кода
- Усложняет дебаг
- Методы с динамической сигнатурой — ЗЛО
- Ограничивают функциональность IDE



Магия

- Усложняет понимание кода
 - Усложняет дебаг
 - Методы с динамической сигнатурой — ЗЛО
 - Ограничивают функциональность IDE
-
- К вам не прилетит сова с письмом из Хогвартса



Автокомплит в IDE

- Постоянно открыта документация
- IDE не подсказывает
- Самая частая функция в работе — поиск по всему проекту
- Проблемы с использованием рефакторинга в PhpStorm
- Непонятная документация

```
$role = $request->input('role');

$users = DB::table('users')
    ->when($role, function ($query, $role) {
        return $query->where('role_id', $role);
    })
    ->get();
```

Что такое \$query!?

Отсутствие прекомпиляции

- Очень много тяжелых операций в рантайме
- Тяжелый роутинг из-за RegExr

Как жить-то с ЭТИМ

- Отделять бизнес-логику от фреймворка
- Проводить ревью всего кода
- Не допускать протечки абстракций
- Не слушать миллениалов, которые говорят, что в Laravel не нужно использовать паттерн репозиторий
- Писать много кода

Спасибо за внимание!



ALEXANDER_GORSHKOV



@PHPEOL



<https://mznx.ru>

