

# Symfony2を1年間実践で 使ってみて学んだこと

株式会社クロコス  
小川雄大

# 自己紹介

- 小川 雄大 / おがわ かつひろ @fivestr
- 株式会社クロコス
- 日本Symfonyユーザー会
- PHP歴6年 (≡ プログラミング歴)
- <http://fivestar.hatenablog.com/>



# 著書

- パーフェクトPHP



- symfony 1.4によるWebアプリケーション開発
- 日経ソフトウェア

# 今日の内容

- クロコスの活動紹介
- Symfonyについて
- この1年で学んだこと
- 書籍の宣伝（一番大事）

# 株式会社クロコス

- 2011.02.07に創立
  - <http://crococos.co.jp/>



- 5人のPHPエンジニア



- 主な業務はFacebookアプリ開発
  - 受託ではなくASP

# サービス紹介(企業向け)

- Crocosマーケティング
  - Facebook向けマーケティングスイート
  - 現在8つのアプリを提供
- Crocosカタログ
  - 楽天上で販売している商品をFacebookページ上でも表示するアプリ

# サービス紹介(一般向け)

- Crocosスケジューラ
- Crocosスポット
- Crocosよせがき
- Crocos懸賞
- すべてFacebookアプリ

# Facebook公認！

- クロコスがFacebookの認定マーケティングデベロッパーになりました！（日本初！）
- 4.19に発表されたばかり
- 国内では4社のみ



# 開発のスタイル

- 各アプリごとに担当が1人
- 2人1チームで進捗やタスクを管理
- 各自、1日～3週間くらいで機能を作ってリリースを繰り返している

# 開発について

- サーバサイドはすべてPHP
- Symfony2フレームワークを採用

# Symfony2

- 比較的新しいフレームワーク
- PHP 5.3以上



Symfony

# Symfony2を採用した経緯

- 新しかったから.....
  - 採用した当時はまだアルファ版
- PHP 5.3向けのフレームワークとして他にまともなものがない

# Symfonyの特徴

- Symfonyは高い柔軟性を持つ
- 設計の自由度が高いため、設計スキルがそれなりに求められる
- 開発初心者には不向き

# Symfonyとモデル

- Symfonyでは様々なことが**モデルを中心**に行われる
  - 入力フォームを扱う処理の流れ
    1. フォームデータをモデルに反映
    2. モデルの状態をバリデーション
    3. モデルのデータをデータベースへ保存

# Symfonyとモデル

- SymfonyではモデルはピュアなPHPオブジェクトとして扱う
- やりたいことに最適化したオブジェクトを用意するとスムーズに問題を解決可能

# Symfonyに限らず

- モデルはアプリケーションでもっとも重要な部分
- フレームワークが用意したクラスだけでなく、必要にあわせて柔軟に実装する



# Symfonyとバンドル

- Symfonyの機能はバンドルという単位でまとめられており、必要なものだけ使うことができる
- フレームワークが用意する機能が合わなければ無理に使う必要はない

# チームにあったものを使う

- Symfonyを使い始めて1番使いづらかったのが標準の認証の仕組み
- 設定が複雑で扱いづらかったので、シンプルな認証の仕組みを実装
- <https://github.com/crocos/CrocosSecurityBundle>

# バンドルによる再利用

- 使い回せるコードはバンドルにまとめて使い回している
- Gitで管理している

# シンプルに作ること

- どんなに柔軟にできるからとはいえ、複雑化しては管理が大変
- 必要な分をシンプルに実装していくことが効率的にアプリを開発していくコツ

# YAGNI原則

- “You ain’t gonna need it” の略
- 必要になるまで実装しない

# DI

- “Dependency Injection” の略
- 依存性の注入という意味
- オブジェクト間の依存度を抑えるためのテクニック

```
// not DI
public function __construct()
{
    $this->logger = new Logger();
}

// DI!!
public function __construct(Logger $logger)
{
    $this->logger = $logger;
}
```

# DIコンテナ

- 依存関係を設定しておくこと、うまいことやってくれるオブジェクト
- 使いすぎると複雑になるのでほどほどにしておく



# 最近の課題

- アプリのパフォーマンスが悪い
  - そもそもチューニングをしていない
  - Symfony自体もわりと重い
- テストを書く
- PHP 5.4化

# Symfonyってどうなの？

- 僕みたいな脳みそオブジェクト指向でできている人に適している
- デザインパターンなどを随所で取り入れているため、パターンを知っているとわりとスムーズに学習できる
- ドメイン駆動設計 (DDD)

# 書籍紹介

- 僕がこれまで開発を行ってきた中で学んだことを書籍にしました
- 現場の開発者向け

# 効率的なWeb アプリケーション の作り方

- 技術評論社
- 2,780円
- 352ページ
- 5.29発売

ISBN978-4-7741-5082-6

効率的なWebアプリケーションの作り方  
—PHPによるモダン開発入門—

## 効率的な Webアプリケーション の作り方

—PHPによるモダン開発入門—

小川 雄大

プロの現場を体験してみませんか?  
最新PHPプログラミングのすべてがわかる

- MVCにそったクリーンなコード
- フレームワークを活用した開発スタイル
- オブジェクト指向の基本と応用
- Gitによるバージョン管理.....

小川  
雄大



How to make efficient Web applications  
Introduction to modern development with PHP.

技術評論社

# Part I MVC開発の基礎知識

- 1章 MVC開発の概要
- 2章 オブジェクト指向

## Part2 フレームワークを利用 する利点

- 3章 レガシーコードの欠点
- 4章 リファクタリング・デザインパターン  
実践
- 5章 フレームワークを活用する

# Part3 モダンPHP Webアプリケーション開発

- 6章 アプリケーション開発を始めるための準備
- 7章 データベースとエンティティ
- 8章 コントローラとビュー
- 9章 フォームとデータベース

# Part4 機能の作成と実装のパターン化

- 10章 ユーザ登録
- 11章 イテレーション
- 12章 管理画面
- 13章 コマンドラインスクリプト
- 14章 ユニットテスト
- 15章 機能テスト



How to make efficient Web applications  
Introduction to modern development with PHP.

# 効率的な Webアプリケーション の作り方

—PHPによるモダン開発入門—

小川 雄大



プロの現場を体験してみませんか?  
最新PHPプログラミングのすべてがわかる

- MVCにそったクリーンなコード
- フレームワークを活用した開発スタイル
- オブジェクト指向の基本と応用
- Gitによるバージョン管理.....

技術評論社

効率的なWebアプリケーションの作り方  
—PHPによるモダン開発入門—

How to make efficient Web applications  
Introduction to modern development with PHP.

小川 雄大

