

Seni projektai, nauji įrankiai

VilniusPHP Susitikimas #1

Povilas Balzaravičius

2012 m. gruodžio 6 d.

Kas aš toks?

- Povilas Balzaravičius
- @Pawka
- github.com/pawka
- linkedin.com/in/pawka
- pawka.linija.net



Seni projektai

Kas jie tokie?

- PHP 4.x/5.x.
- Kodas \geq 4 metų senumo.
- Niekur nematytas kodo stilius(-ai).
- Nenaudojamas žmonijai žinomas karkasas.
- **include**, **require** ir draugai.

Kylančios problemos

- Didžiulės sąnaudos tvarkingai perrašyti kodą.
- Naujo funkcionalumo pridėjimas reikalauja daug laiko.
- Kažką pajudinus viskas griūna.
- Šlykštoka dirbti...

Pradedam!

Stilius

Naujas projektas - naujas programavimo stilius.

Standartas

PSR

PSR Standartas

PHP Specification Request - programavimo stiliaus rekomendacija.

PSR Standartas

PHP Specification Request - programavimo stiliaus rekomendacija.

Sudaro trys dokumentai:

PSR-0 Autoload standartas.

PSR Standartas

PHP Specification Request - programavimo stiliaus rekomendacija.

Sudaro trys dokumentai:

PSR-0 Autoload standartas.

PSR-1 "Basic Coding Standard" - Koduotė, PHP *tagai*, konstantos, klasių ir metodų pavadinimai, ...

PSR Standartas

PHP Specification Request - programavimo stiliaus rekomendacija.

Sudaro trys dokumentai:

PSR-0 Autoload standartas.

PSR-1 “Basic Coding Standard” - Koduotė, PHP *tagai*, konstantos, klasių ir metodų pavadinimai, ...

PSR-2 “Coding Style Guide” - Tarpai, skliausteliai ir kableliai :-)

PSR Standartas

PHP Specification Request - programavimo stiliaus rekomendacija.

Sudaro trys dokumentai:

PSR-0 Autoload standartas.

PSR-1 “Basic Coding Standard” - Koduotė, PHP *tagai*, konstantos, klasių ir metodų pavadinimai, ...

PSR-2 “Coding Style Guide” - Tarpai, skliausteliai ir kableliai :-)

Standarto aprašymas: github.com/php-fig/fig-standards/

Sprendimas

php-cs-fixer

PHP Coding Standards Fixer

Įrankis skirtas kodo stiliaus tvarkymui pagal **PSR-1** ir **PSR-2** standartus.

- Autorius: Fabien Potencier
- <http://cs.sensiolabs.org/>
- Galimybė tvarkyti tik tam tikras sritis (identacija, skliaustų išdėstymas, ...)
- ...ir/arba naudoti paruoštas konfigūracijas (sf20, sf21, magento, default).

PHP Coding Standards Fixer - naudojimas

- 1 `sudo wget http://cs.sensiolabs.org/get/php-cs-fixer.phar -O /usr/local/bin/php-cs-fixer`
- 2 `sudo chmod a+x /usr/local/bin/php-cs-fixer`
- 3 `php php-cs-fixer.phar fix /path/to/dir`
- 4 Džiaugiamės tvarkingu kodu :-)

PHP Coding Standards Fixer - naudojimas

- 1 `sudo wget http://cs.sensiolabs.org/get/php-cs-fixer.phar -O /usr/local/bin/php-cs-fixer`
- 2 `sudo chmod a+x /usr/local/bin/php-cs-fixer`
- 3 `php php-cs-fixer.phar fix /path/to/dir`
- 4 Džiaugiamės tvarkingu kodu :-) (dažniausiai)

Išorinės bibliotekos

Naudojam tai, kas jau sukurta.

Problemos

- Nepatogu įtrauktį į kodą.
- Gali priklausyti nuo kitų bibliotekų.
- Saugomos kartu su projekto kodu.
- Nepatogu atnaujinti.

Sprendimas

Composer

Composer

- Autorius: Jordi Boggiano
- <http://getcomposer.org/>
- Suranda, instaliuoja, atnaujina ir *autoloadina* paketus.

Composer - naudojimas

- 1 `curl -s https://getcomposer.org/installer | php`
- 2 Parsiunčiamas **composer.phar**
- 3 `php composer.phar <komanda>`

Composer - paketų instaliavimas

- 1 Susirandam paketą ir jo versiją:
`php composer.phar search kažkas` arba `packagist.org`.
- 2 Pvz. **doctrine/dbal 2.3.0**.
- 3 Įtraukiam paketą į **composer.json**.
- 4 `php composer.phar install` (arba `update`).
- 5 Parsiūs paketus ir sugeneruos Autoload failus.

```
pawka@yoda (192.168.2.5) - byobu
pawka@yoda ~-> php composer.phar
Composer version eef64d1

Usage:
  [options] command [arguments]

Options:
  --help           -h Display this help message.
  --quiet          -q Do not output any message.
  --verbose        -v Increase verbosity of messages.
  --version        -V Display this application version.
  --ansi           Force ANSI output.
  --no-ansi        Disable ANSI output.
  --no-interaction -n Do not ask any interactive question.
  --profile        Display timing and memory usage information
  --working-dir    -d If specified, use the given directory as working directory.

Available commands:
  about           Short information about Composer
  config          Set config options
  create-project  Create new project from a package into given directory.
  depends         Shows which packages depend on the given package
  dump-autoload   Dumps the autoloader
  dumpautoload    Dumps the autoloader
  help            Displays help for a command
  init            Creates a basic composer.json file in current directory.
  install         Installs the project dependencies from the composer.lock file if present, or falls back on the composer.json.
  list            Lists commands
  require         Adds required packages to your composer.json and installs them
  search          Search for packages
  self-update     Updates composer.phar to the latest version.
  selfupdate      Updates composer.phar to the latest version.
  show            Show information about packages
  status          Show a list of locally modified packages
  update          Updates your dependencies to the latest version according to composer.json, and updates the composer.lock file.
  validate        Validates a composer.json

pawka@yoda ~-> █
1*$ 2-@ $ irssi
U 12.04.1
pawka@yoda 192.168.20.64 Menu:<F9>
v445kb 4d1h 100%+ 72.2Mbps94% 0.57 4x2.4GHz 3.8GB69% 2012-12-06 17:38:44
```


Composer - naudojimas

composer.json pavyzdys

```
1 {  
2     "name" : "Project/name",  
3     "require" : {  
4         "doctrine/dbal" : "v2.3.*",  
5         "monolog/monolog" : "1.2.*"  
6     },  
7     "minimum-stability" : "dev"  
8 }  
9
```

Composer - naudojimas

composer.json pavyzdys

```
1 {  
2     "name" : "Project/name",  
3     "require" : {  
4         "doctrine/dbal" : "v2.3.*",  
5         "monolog/monolog" : "1.2.*"  
6     },  
7     "minimum-stability" : "dev"  
8 }  
9
```

O tada...

```
1 <?php  
2  
3 require_once(__DIR__ . '/vendor/autoload.php');  
4  
5
```

Iš ko rinktis?

Symfony Components

`symfony.com/components`

- BrowserKit
- ClassLoader
- Config
- Console
- CssSelector
- DependencyInjection
- DomCrawler
- EventDispatcher
- Finder
- Form
- HttpFoundation
- HttpKernel
- Locale
- Process
- Routing
- Security
- Serializer
- Templating
- Translation
- Validator
- Yaml

Zend Framework 2

48 atskiri komponentai

`framework.zend.com`

Kaip dirbu aš?

Darbo planas

- 1 Stiliaus sutvarkymas.
- 2 Composer paruošimas.
- 3 DBAL (vienareikšmiškai **doctrine/dbal**).
- 4 Dependency Injection Container.
- 5 *Routing*'o įdiegimas.
- 6 Naujo kodo rašymas teisinga tvarka.

Dependency Injection Container

Kam reikalingas?

- Objektų kūrimo taisyklės saugomos vienoje vietoje.
- *Lazy Loading* - objektai kuriami tik tada, kai jų prireikia.

Fabien Potencier

... when you need to manage a lot of different objects with a lot of dependencies, a Dependency Injection Container can be really helpful (think of a framework for instance).

Pasiskaitymui: “What is Dependency Injection” by @fabpot.

Pimple

A simple Dependency Injection Container for PHP 5.3

- Autorius: Fabien Potencier
- `pimple.sensiolabs.org`
- Paprastas
- Greitas
- Integravimas - kelios eilutės.
- Primityvus (pvz. lyginant su Symfony DependencyInjection)

Routing

Kam reikalingas?

- Iškviešti reikiamą *Controller'io Actions'ą* pagal URL.
- Iškviečiant priskirti kintamuosius iš URL.
- URL generatorius pagal *route'o* pavadinimą.
- Dažnai senesni projektai neturi :-)

Symfony Routing Component

- `symfony.com/doc/current/components/routing/`
- Integravimas - kelios eilutės.
- Sprendžia visas anksčiau aprašytas problemas.

Teisingas kodas

Service-oriented architecture

Symfony Glossary

A Service is a generic term for any PHP object that performs a specific task. A service is usually used "globally", such as a database connection object or an object that delivers email messages. In Symfony2, services are often configured and retrieved from the service container. An application that has many decoupled services is said to follow a service-oriented architecture.

SOA Robotikos taisyklės

- Modeliai - žemiausias lygmuo, vieni apie kitus nežino.
- Servisas žino apie jam priklausančius modelius.
- Servisas gali žinoti apie kitus servisus.
- Kontroleriai nežino nieko apie modelius ir naudoja TIK servisus.

Apibendrinimas

- Naują kodą rašykite tvarkingai.
- Seną kodą perrašinėkite nedidelėmis dalimis ir tik tada, kai to reikia.
- Naudokite jau sukurtus įrankius.
- Sugaišit laiko sutvarkymui, bet laimėsit kurdami naujas funkcijas.

Ačiū