

## Program szkolenia:

# Nowoczesne narzędzia developera PHP

### Informacje:

<b>Nazwa:</b>	<b>Nowoczesne narzędzia developera PHP</b>
<b>Kod:</b>	<b>PHP-tools</b>
<b>Kategoria:</b>	PHP
<b>Odbiorcy:</b>	developerzy
<b>Czas trwania:</b>	2 dni
<b>Forma:</b>	100% wykłady

---

Szkolenie zostało przygotowane z myślą o programistach PHP, w celu zapoznania ich z zestawem obecnie wykorzystywanych narzędzi przy implementacji aplikacji w języku PHP.

### Zalety szkolenia:

- Najlepsze narzędzia sprawdzone przez wieloletnich praktyków
- Kruczki i najlepsze praktyki

## Szczegółowy program:

### 1. Wprowadzenie

1.1. Krótki rys historyczny języka PHP

1.2. PHP 4 vs wersje PHP 5

1.3. Ewolucja narzędzi developerskich

### 2. Wirtualizacja środowiska developerskiego z Vagrant

2.1. Koncepcja wirtualizacji środowiska

2.2. Instalacja oprogramowania i pierwszej maszyny wirtualnej

2.3. Zarządzanie Vagrantem z poziomu linii poleceń

2.4. Opis konfiguracji maszyny wirtualnej poprzez plik Vagrantfile

2.5. Prowizjonowanie maszyny wirtualnej

2.6. Konfiguracja sieci

2.7. Środowiska wieloserwerowe

2.8. Providers

### 3. Zarządzanie dependencjami aplikacji z Composer

3.1. Jaki problem chcemy rozwiązać?

3.2. Koncepcja zarządzania dependencjami aplikacji

3.3. Instalacja Composera

3.4. Zarządzanie Composerem z poziomu linii poleceń

3.5. Struktura pliku composer.json

3.6. Definiowanie zależności projektowych

3.7. Definiowanie autoloaderów

3.8. Zamrażanie zależności poprzez lock file

3.9. Wykonywanie skryptów przed i/lub po instalacji pakietu

## 3.10. Repozytorium Packagist

**4. Rekomendowane propozycje standaryzacji PHP (PSR)**

4.1. Standard PSR-1: Basic Coding Standard

4.2. Standard PSR-2: Coding Style Guide

4.3. Standard PSR-3: Logger Interface

4.4. Standard PSR-4: Improved Autoloading

**5. Nowoczesne frameworki PHP**

5.1. Przegląd możliwości i funkcjonalności popularnych frameworków PHP

5.1.1. Symfony 2

5.1.2. Yii

5.1.3. Zend

5.1.4. Laravel

5.1.5. Phalcon

**6. Techniki testowania oprogramowania**

6.1. Test Driven Development

6.2. Behavior Driven Development

6.3. Przegląd popularnych narzędzi do testowania oprogramowania

6.3.1. PHPUnit

6.3.2. SimpleTest

6.3.3. Mink

6.3.4. Codeception

6.3.5. Behat

6.3.6. PHPSpec

6.3.7. Mockery

6.3.8. Prophecy

## 7. Buildowanie oraz deployment aplikacji

7.1. Wprowadzenie do koncepcji buildów automatycznego deploymentu aplikacji

7.1.1. Zarządzanie zależnościami

7.1.2. Kompilacja, minifikacja oraz obfuskacja assetów JS/CSS

7.1.3. Uruchomianie testów

7.1.4. Tworzenie artefaktów

7.1.5. Deployment

7.2. Continuous Integration

7.3. Narzędzia

7.3.1. Phing

7.3.2. Capifony

7.3.3. Jenkins

7.3.4. PHPCI

## 8. Debuggowanie i profilowanie aplikacji PHP

8.1. Wprowadzenie do tematyki

8.2. Profilowanie wydajnościowe aplikacji

8.3. Narzędzia

8.3.1. XHProf

8.3.2. Xdebug

8.3.3. Analiza uzyskanych informacji w aplikacji kCacheGrind

## 9. Logowanie błędów i pracy aplikacji

9.1. Wprowadzenie do tematyki

9.2. Monolog jako wzorcowa implementacja standardu PSR-3

9.3. Monolog jako interfejs do wielu backendów składających dane

9.4. Eksport logów do innych systemów, w tym analityki i big-data

## 9.5. Narzędzia

9.5.1. NewRelic

9.5.2. Loggly

9.5.3. Logstash

9.5.4. Kibana

9.5.5. ElasticSearch

9.5.6. Fluentd