

Spis treści

<b>O autorze</b>	<b>15</b>
<b>O recenzencie technicznym</b>	<b>17</b>
<b>Podziękowania</b>	<b>19</b>
<b>Wprowadzenie</b>	<b>21</b>
<b>Część I. Obiekty</b>	<b>23</b>
<b>Rozdział 1. PHP – projektowanie i zarządzanie</b>	<b>25</b>
Problem	25
PHP a inne języki programowania	26
O książce	28
Obiekty	28
Wzorce	28
Narzędzia	29
Nowości w piątym wydaniu	30
Podsumowanie	30
<b>Rozdział 2. PHP i obiekty</b>	<b>31</b>
Nieoczekiwany sukces obiektów w PHP	31
PHP/FI – u zarania języka	31
PHP3 – składniowy lukier	31
Cicha rewolucja – PHP4	32
PHP5 – nieuchronne zmiany	33
PHP7 – doganianie reszty świata	34
Debata obiektowa – za czy przeciw?	34
Podsumowanie	35
<b>Rozdział 3. Obiektowy elementarz</b>	<b>37</b>
Klasy i obiekty	37
Pierwsza klasa	37
Pierwszy obiekt (lub dwa)	38
Definiowanie składowych klasy	39
Metody	41
Metoda konstrukcji obiektu	42
Typy argumentów metod	43
Typy elementarne	44
Typy obiektowe	47
Dziedziczenie	50

Problemy związane z dziedziczeniem	51
Stosowanie dziedziczenia	55
Zarządzanie dostępem do klasy — słowa public, private i protected	60
Podsumowanie	64
<b>Rozdział 4. Zaawansowana obsługa obiektów</b>	<b>65</b>
Metody i składowe statyczne	65
Składowe stałe	68
Klasy abstrakcyjne	69
Interfejsy	71
Cechy typowe	73
Zadanie dla cech typowych	73
Definiowanie i stosowanie cechy typowej	74
Stosowanie wielu cech typowych	75
Łączenie cech z interfejsami	75
Unikanie kolizji nazw metod za pomocą słowa insteadof	76
Aliasy metod cech typowych	77
Cechy typowe z metodami statycznymi	78
Dostęp do składowych klasy włączającej	79
Definiowanie metody abstrakcyjnej cechy typowej	79
Zmiana dostępności metod cech typowych	80
Późne wiązanie statyczne: słowo static	81
Obsługa błędów	84
Wyjątki	85
Klasy i metody finalne	92
Klasa do obsługi błędów wewnętrznych	93
Przechwytywanie chybionych wywołań	94
Definiowanie destruktorów	99
Wykonywanie kopii obiektów	100
Reprezentacja obiektu w ciągach znaków	103
Wywołania zwrotne, funkcje anonimowe i domknięcia	104
Klasy anonimowe	108
Podsumowanie	109
<b>Rozdział 5. Narzędzia obiektowe</b>	<b>111</b>
PHP a pakiety	111
Pakiety i przestrzenie nazw w PHP	111
Automatyczne wczytywanie kodu	119
Klasy i funkcje pomocnicze	122
Szukanie klasy	123
Badanie obiektów i klas	124
Pozyskiwanie ciągu pełnej nazwy klasy	125
Badanie metod	126
Badanie składowych	127
Badanie relacji dziedziczenia	127
Badanie wywołań metod	128
Interfejs retrospekcji — Reflection API	129
Zaczynamy	129

Pora zakasać rękawy	130
Badanie klasy	132
Badanie metod	133
Badanie argumentów metod	135
Korzystanie z retrospekcji	136
Podsumowanie	139
<b>Rozdział 6. Obiekty a projektowanie obiektowe</b>	<b>141</b>
Czym jest projektowanie?	141
Programowanie obiektowe i proceduralne	142
Odpowiedzialność	145
Spójność	146
Sprzęganie	146
Ortogonalność	146
Zasięg klas	147
Polimorfizm	148
Hermetyzacja	149
Nieważne jak	150
Cztery drogowskazy	151
Zwielokrotnianie kodu	151
Przemądrzałe klasy	151
Złota rączka	151
Za dużo warunków	151
Język UML	152
Diagramy klas	152
Diagramy sekwencji	157
Podsumowanie	159
<b>Część II. Wzorce</b>	<b>161</b>
<b>Rozdział 7. Czym są wzorce projektowe? Do czego się przydają?</b>	<b>163</b>
Czym są wzorce projektowe?	163
Wzorzec projektowy	165
Nazwa	165
Problem	166
Rozwiązanie	166
Konsekwencje	166
Format wzorca według Bandy Czworga	166
Po co nam wzorce projektowe?	167
Wzorzec projektowy definiuje problem	167
Wzorzec projektowy definiuje rozwiązanie	167
Wzorce projektowe są niezależne od języka programowania	167
Wzorce definiują słownictwo	168
Wzorce są wypróbowane	168
Wzorce mają współpracować	169
Wzorce promują zasady projektowe	169
Wzorce są stosowane w popularnych frameworkach	169
Wzorce projektowe a PHP	169

Podsumowanie	169
<b>Rozdział 8. Wybrane zasady wzorców</b>	<b>171</b>
Olśnienie wzorcami	171
Kompozycja i dziedziczenie	172
Problem	172
Zastosowanie kompozycji	175
Rozprężanie	177
Problem	178
Oslabianie sprzężenia	179
Kod ma używać interfejsów, nie implementacji	180
Zmienne koncepcje	181
Nadmiar wzorców	182
Wzorce	182
Wzorce generowania obiektów	183
Wzorce organizacji obiektów i klas	183
Wzorce zadaniowe	183
Wzorce korporacyjne	183
Wzorce baz danych	183
Podsumowanie	183
<b>Rozdział 9. Generowanie obiektów</b>	<b>185</b>
Generowanie obiektów — problemy i rozwiązania	185
Wzorzec Singleton	189
Problem	189
Implementacja	190
Konsekwencje	192
Wzorzec Factory Method	192
Problem	192
Implementacja	195
Konsekwencje	196
Wzorzec Abstract Factory	197
Problem	197
Implementacja	198
Konsekwencje	200
Prototyp	201
Problem	202
Implementacja	202
Naginanie rzeczywistości — wzorzec Service Locator	205
Doskonała izolacja — wstrzykiwanie zależności	206
Problem	207
Implementacja	207
Konsekwencje	210
Podsumowanie	210
<b>Rozdział 10. Wzorce elastycznego programowania obiektowego</b>	<b>211</b>
Strukturalizacja klas pod kątem elastyczności obiektów	211
Wzorzec Composite	211

Problem	212
Implementacja	214
Konsekwencje	217
Composite — podsumowanie	220
Wzorzec Decorator	220
Problem	221
Implementacja	223
Konsekwencje	226
Wzorzec Facade	226
Problem	227
Implementacja	228
Konsekwencje	229
Podsumowanie	229
<b>Rozdział 11. Reprezentacja i realizacja zadań</b>	<b>231</b>
Wzorzec Interpreter	231
Problem	231
Implementacja	232
Ciemne strony wzorca Interpreter	239
Wzorzec Strategy	240
Problem	240
Implementacja	241
Wzorzec Observer	244
Implementacja	246
Wzorzec Visitor	252
Problem	252
Implementacja	253
Wady wzorca Visitor	258
Wzorzec Command	258
Problem	258
Implementacja	259
Wzorzec Null Object	262
Problem	263
Implementacja	265
Podsumowanie	266
<b>Rozdział 12. Wzorce korporacyjne</b>	<b>267</b>
Przeгляд architektury	267
Wzorce	268
Aplikacje i warstwy	268
Małe oszustwo na samym początku	270
Wzorzec Registry	271
Implementacja	272
Warstwa prezentacji	275
Wzorzec Front Controller	276
Wzorzec Application Controller	285
Zarządzanie danymi w komponentach	290
Wzorzec Page Controller	296

Wzorce Template View i View Helper	300
Warstwa logiki biznesowej	302
Wzorzec Transaction Script	303
Wzorzec Domain Model	306
Podsumowanie	309
<b>Rozdział 13. Wzorce bazodanowe</b>	<b>311</b>
Warstwa danych	311
Wzorzec Data Mapper	312
Problem	312
Implementacja	312
Wzorzec Identity Map	325
Problem	325
Implementacja	325
Konsekwencje	328
Wzorzec Unit of Work	328
Problem	328
Implementacja	329
Konsekwencje	332
Wzorzec Lazy Load	332
Problem	332
Implementacja	333
Konsekwencje	334
Wzorzec Domain Object Factory	334
Problem	335
Implementacja	335
Konsekwencje	336
Wzorzec Identity Object	337
Problem	338
Implementacja	338
Konsekwencje	343
Wzorce Selection Factory i Update Factory	343
Problem	343
Implementacja	344
Konsekwencje	347
Co zostało z wzorca Data Mapper?	347
Podsumowanie	349
<b>Część III. Narzędzia</b>	<b>351</b>
<b>Rozdział 14. Dobre (i złe) praktyki</b>	<b>353</b>
Nie tylko kod	353
Pukanie do otwartych drzwi	354
Jak to zgrać?	355
Uskrzydlenie kodu	356
Standardy	357
Vagrant	358
Testowanie	358

Ciągła integracja	359
Podsumowanie	359
<b>Rozdział 15. Standardy PHP</b>	<b>361</b>
Po co te standardy?	361
Jakie są standardowe rekomendacje PHP?	362
Dlaczego akurat PSR?	362
Rekomendacje PSR — dla kogo?	363
Kodowanie z klasą	363
Podstawowy standard kodowania PSR-1	364
Rekomendacja stylu kodowania PSR-2	366
Sprawdzanie i poprawianie kodu	368
PSR-4 — automatyczne ładowanie	370
Zasady, które są dla nas ważne	370
Podsumowanie	373
<b>Rozdział 16. Używanie i tworzenie komponentów PHP za pomocą Composer</b>	<b>375</b>
Czym jest Composer?	375
Instalowanie Composer	376
Instalowanie (zbioru) pakietów	376
Instalowanie pakietu z poziomu wiersza poleceń	377
Wersje	377
Element require-dev	378
Composer i automatyczne ładowanie kodu	379
Tworzenie własnego pakietu	380
Dodawanie informacji o pakiecie	380
Pakiety systemowe	380
Dystrybucja za pośrednictwem repozytorium Packagist	381
Odrobina prywatności	384
Podsumowanie	385
<b>Rozdział 17. Zarządzanie wersjami projektu z systemem Git</b>	<b>387</b>
Po co mi kontrola wersji?	387
Skąd wziąć klienta Git?	388
Obsługa repozytorium Git online	389
Konfigurowanie serwera Git	391
Tworzenie repozytorium zdalnego	391
Rozpoczynamy projekt	393
Klonowanie repozytorium	395
Wprowadzanie i zatwierdzanie zmian	396
Dodawanie i usuwanie plików i katalogów	399
Dodawanie pliku	399
Usuwanie pliku	399
Dodawanie katalogu	400
Usuwanie katalogów	400
Etykietowanie wersji	400
Rozgałęzianie projektu	401

Podsumowanie	407
<b>Rozdział 18. Testy jednostkowe z PHPUnit</b>	<b>409</b>
Testy funkcjonalne i testy jednostkowe	409
Testowanie ręczne	410
PHPUnit	412
Tworzenie przypadku testowego	412
Metody asercji	414
Testowanie wyjątków	415
Uruchamianie zestawów testów	416
Ograniczenia	417
Atrapy i imitacje	419
Dobry test to obłany test	421
Testy dla aplikacji WWW	423
Przygotowanie aplikacji WWW do testów	424
Proste testy aplikacji WWW	426
Selenium	427
Słowo ostrzeżenia	432
Podsumowanie	434
<b>Rozdział 19. Automatyzacja instalacji z Phingiem</b>	<b>435</b>
Czym jest Phing?	436
Pobieranie i instalacja pakietu Phing	436
Montowanie dokumentu kompilacji	437
Różnicowanie zadań kompilacji	438
Właściwości	440
Typy	446
Operacje	450
Podsumowanie	454
<b>Rozdział 20. Vagrant</b>	<b>455</b>
Problem	455
Odrobina przygotowań	456
Wybór i instalacja środowiska Vagrant	456
Montowanie lokalnych katalogów w maszynie wirtualnej Vagranta	458
Zaopatrywanie maszyny wirtualnej	459
Konfigurowanie serwera WWW	460
Konfigurowanie MySQL	461
Konfigurowanie nazwy hosta	462
Kilka słów na koniec	463
Podsumowanie	464
<b>Rozdział 21. Ciągła integracja kodu</b>	<b>465</b>
Czym jest ciągła integracja?	465
Przygotowanie projektu do ciągłej integracji	467
Instalowanie Jenkinsa	475
Instalowanie rozszerzeń Jenkinsa	475
Konfigurowanie klucza publicznego serwera Git	477



Instalowanie projektu	478
Pierwsza kompilacja	479
Konfigurowanie raportów	482
Automatyzacja kompilacji	484
Podsumowanie	486
<b>Rozdział 22. Obiekty, wzorce, narzędzia</b>	<b>487</b>
Obiekty	487
Wybór	488
Hermetyzacja i delegowanie	488
Oslabianie sprzężenia	488
Zdatność do wielokrotnego stosowania kodu	489
Estetyka	489
Wzorce	490
Co dają nam wzorce?	490
Wzorce a zasady projektowe	491
Narzędzia	492
Testowanie	493
Standardy	493
Zarządzanie wersjami	493
Automatyczna kompilacja (instalacja)	494
System integracji ciągłej	494
Co pominęliśmy?	494
Podsumowanie	495
<b>Dodatki</b>	<b>497</b>
<b>Dodatek A Bibliografia</b>	<b>499</b>
Książki	499
Publikacje	500
Witryny WWW	500
<b>Dodatek B Prosty analizator leksykalny</b>	<b>503</b>
Skaner	503
Analizator leksykalny	510
<b>Skorowidz</b>	<b>523</b>