

spis treści

przedmowa xv

podziękowania xvii

o książce	xix
o autorach	xxv
o ilustracji na okładce	xxvii
Część 1 Umiejscowienie wstrzykiwania zależności na mapie	1
1 Podstawy Wstrzykiwania zależności: co, jak i dlaczego	3
1.1. Pisanie utrzymywalnego kodu	5
Powszechne mity na temat DI	6
Rozumienie celu DI	9
1.2. Prosty przykład: Hello DI!	16
Kod Hello DI!	16
Korzyści (płynące) z DI	19
1.3. Co wstrzykiwać, a czego nie wstrzykiwać	28
STABILNE ZALEŻNOŚCI	29
NIESTABILNE ZALEŻNOŚCI	30
1.4. Zakres DI	31
KOMPOZYCJA OBIEKTOWA	32
CYKL ŻYCIA OBIEKTU	33
PRZECHWY-TYWANIE	34
DI w trzech wymiarach	35
1.5. Konkluzje	36

2 Pisanie ściśle powiązanego kodu	39
2.1. Budowanie ściśle powiązanej aplikacji	40
Poznajmy Mary Rowan	41
Tworzenie warstwy danych	42
Tworzenie warstwy domeny	44
Tworzenie warstwy interfejsu użytkownika	48
2.2. Ocena aplikacji ściśle powiązanej	50
Ocena grafu zależności	50
Ocena tworzenia strukturalności	50
2.3. Analiza braku strukturalności	53
Analiza wykresu zależności	53
Analiza interfejsu dostępu do danych	54
Inne kwestie	56
2.4. Konkluzja	57
3 Pisanie luźno powiązanego kodu	59
3.1. Ponowne budowanie aplikacji e-commerce	60
Budowanie bardziej utrzymywanego UI	63
Budowanie niezależnego modelu domeny	69
Budowanie nowej warstwy dostępu do danych	78
Implementacja Adaptera IUserContext specyficznego dla ASP.NET Core	79
Tworzenie aplikacji w PODSTAWIE KOMPOZYCJI	82
3.2. Analiza luźno powiązanej implementacji	82
Zrozumienie interakcji między komponentami	83
Analizowanie nowego grafu zależności	84
Część 2 katalog	89
4 Wzorce DI	91

4.1. PODSTAWA KOMPOZYCJI	93
Jak działa PODSTAWA KOMPOZYCJI	95
Użycie KONTENERA DI w PODSTAWIE KOMPOZYCJI	96
Przykład: Implementowanie PODSTAWY KOMPOZYCJI przy użyciu CZYSTEGO DI	97
Pozorna eksplozja zależności	100
4.2. WSTRZYKIWANIE KONSTRUKTOREM	102
Jak działa WSTRZYKIWANIE KONSTRUKTOREM	102
Kiedy używać WSTRZYKIWANIA KONSTRUKTOREM	104
Znany sposób użycia WSTRZYKIWANIA KONSTRUKTOREM	107
Przykład: Dodawanie przeliczania walut do wyróżnionych produktów	108
Podsumowanie	110
4.3. WSTRZYKIWANIE METODĄ	112
Jak działa WSTRZYKIWANIE METODĄ	112
Kiedy używać WSTRZYKIWANIA METODĄ	113
Znany sposób użycia WSTRZYKIWANIA METODĄ	119
Przykład: Dodawanie przewalutowania do ENCJI Product	121
4.4. WSTRZYKIWANIE WŁAŚCIWOŚCIĄ	123
Jak działa WSTRZYKIWANIE WŁAŚCIWOŚCIĄ	124
Kiedy używać WSTRZYKIWANIA WŁAŚCIWOŚCIĄ	125
Znane zastosowania WSTRZYKIWANIA WŁAŚCIWOŚCIĄ	128
Przykład: WSTRZYKIWANIE WŁAŚCIWOŚCIĄ jako model rozszerzalności biblioteki do ponownego użycia	128
4.5. Wybieranie wzorca (do pracy)	131
5 Antywzorce DI	135
5.1. ANTYWZORZEC CONTROL FREAK	138

Przykład: CONTROL FREAK przez nowo powstające	
ZALEŻNOŚCI	139
Przykład: CONTROL FREAK przez fabryki	140
Przykład: CONTROL FREAK przez przeciążone konstruktory	146
Analiza CONTROL FREAK	147
5.2. LOKALIZATOR USŁUG	149
Przykład: ProductService przy użyciu LOKALIZATORA USŁUG	151
Analiza LOKALIZATORA USŁUG	154
5.3. KONTEKST OTOCZENIA	158
Przykład: Dostęp do czasu przez KONTEKST OTOCZENIA	159
Przykład: Logowanie przez KONTEKST OTOCZENIA	161
Analiza KONTEKSTU OTOCZENIA	162
5.4. Antywzorzec OGRANICZONEJ KONSTRUKCJI	166
Przykład: Późne wiązanie ProductRepository	167
Analiza OGRANICZONEJ KONSTRUKCJI	169
6 Zapachy kodu	176
6.1. Radzenie sobie z zapachem kodu Przesadnego	
Wstrzykiwania Konstruktozem	177
Rozpoznawanie Przesadnego Wstrzykiwania Konstruktozem	179
Refaktorowanie z Przesadnego Wstrzykiwania Konstruktozem	
do Usług fasadowych	182
Refaktorowanie z Przesadnego Wstrzykiwania Konstruktozem	
do zdarzeń domeny	187
6.2. Nadużywanie Fabryk abstrakcyjnych	195
Nadużywanie Fabryk abstrakcyjnych do pokonania problemów	
cyklu życia	195
Nadużywanie Fabryk abstrakcyjnych do wybierania ZALEŻNOŚCI	

na podstawie danych czasu wykonania	203
6.3. Naprawianie cyklicznych ZALEŻNOŚCI	209
Przykład: Cykliczne ZALEŻNOŚCI spowodowane naruszeniem SRP	211
Analiza cyklu ZALEŻNOŚCI Mary 215	
Refaktoring naruszeń SRP, aby rozwiązać cykl ZALEŻNOŚCI	216
Powszechne strategie przerywania cykli ZALEŻNOŚCI	220
Ostateczność: przerwanie cyklu za pomocą WSTRZYKIWANIA WŁAŚCIWOŚCIĄ	220
Część 3 Czyste Di	225
7 Kompozycja aplikacji	227
7.1. Komponowanie aplikacji konsolowej	230
Przykład: Uaktualnienie kursu walut przy użyciu programu UpdateCurrency	230
Budowanie PODSTAWY KOMPOZYCJI programu UpdateCurrency	231
Komponowanie grafów obiektów w CreateCurrencyParser	232
Głębsze spojrzenie na warstwy UpdateCurrency	233
7.2. Komponowanie aplikacji UWP	234
Kompozycja UWP	235
Przykład: Podłączanie aplikacji typu gruby klient do zarządzania Produktami	235
Implementowanie PODSTAWY KOMPOZYCJI w aplikacji UWP	243
7.3. Komponowanie aplikacji ASP.NET Core MVC	246
Tworzenie własnego aktywatora kontrolera	247
Konstruowanie własnej warstwy middleware przy użyciu CZYSTEGO DI	250

8 Cykl życia obiektu	254
8.1. Zarządzanie CYKLEM ŻYCIA ZALEŻNOŚCI	256
Wprowadzenie ZARZĄDZANIA CYKLEM ŻYCIA	257
Zarządzanie cyklem życia przy użyciu CZYSTEGO DI	261
8.2. Pracowanie z usuwalnymi ZALEŻNOŚCIAMI	264
Konsumowanie usuwalnych ZALEŻNOŚCI	265
Zarządzanie usuwalnymi ZALEŻNOŚCIAMI	269
8.3. Katalog STYLÓW ŻYCIA	275
STYL ŻYCIA SINGLETON	276
PRZEJŚCIOWY STYL ŻYCIA	279
ZAKRESOWY STYL ŻYCIA	281
8.4. Złe wybory STYLU ŻYCIA	287
POJMANE ZALEŻNOŚCI	288
Wybór STYLU ŻYCIA wyciekający do klas konsumujących przez CIEKNAĆCE ABSTRAKCJE	290
Powodowanie błędów współbieżności przez łączenie instancji z cyklem życia jednego wątku	297
9 Przechwytywanie	303
9.1. Przedstawienie PRZECHWYTYWANIA	305
Wzorzec projektowy dekorator	306
Przykład: Implementowanie audytowania przy użyciu Dekoratora	311
9.2. Implementowanie ZAGADNIEŃ PRZEKROJOWYCH	314
Przechwytywanie za pomocą wzorca Bezpiecznik	315
Raportowanie wyjątków przy użyciu wzorca Dekorator	321
Zapobieganie nieautoryzowanemu dostępowi do wrażliwej funkcjonalności przy użyciu Dekoratora	323

10 Rozmyślne Programowanie Aspektowe	326
10.1.Wprowadzenie do AOP	327
10.2.Zasady SOLID	330
ZASADA POJEDYNCZEJ ODPOWIEDZIALNOŚCI (SRP)	331
ZASADA OTWARTE/ZAMKNIĘTE (OCP)	332
ZASADA PODSTAWIENIA LISKOV (LSP)	332
ZASADA SEGREGACJI INTERFEJSÓW (ISP)	333
ZASADA OD-WRÓCENIA ZALEŻNOŚCI (DIP)	333
Zasady SOLID aPRZECHWYTYWANIE	333
10.3.SOLID jako sterownik dla AOP	334
Przykład: Implementowanie funkcjonalności związanych z produktem przy użyciu IProductService	334
Analiza IProductService z perspektywy SOLID	336
Ulepszanie projektu przez zastosowanie zasad SOLID	339
Dodanie większej liczby ZAGADNIENÍ PRZEKROJOWYCH	353
Wnioski	363
11 Programowanie Aspektowe oparte na narzędziach	367
11.1.Dynamiczne PRZECHWYTYWANIE	368
Przykład: PRZECHWYTYWANIE za pomocą Castle Dynamic Proxy	371
Analiza dynamicznego PRZECHWYTYWANIA	373
11.2.Tkanie w czasie kompilacji	375
Przykład: Zastosowanie aspektu transakcji przy użyciu tkania w czasie kompilacji	377
Analiza tkania w czasie kompilacji	379

Część 4 kontenery Di	385
12 Wprowadzenie do Kontenera DI	387
12.1.Wprowadzenie do KONTENERÓW DI	389
Odkrywanie API kontenerów rozwiązującego ZALEŻNOŚCI	389
AUTOMATYCZNE PODŁĄCZANIE	392
Przykład: Implementowanie uproszczonego KONTENERA DI, który wspiera AUTOMATYCZNE PODŁĄCZANIE	393
12.2.Konfigurowanie KONTENERÓW DI	401
Konfigurowanie kontenerów za pomocą plików konfiguracyjnych	403
Kontenery konfiguracyjne przy użyciu KONFIGURACJI JAKO KODU	406
Konfigurowanie kontenerów przez konwencję przy użyciu AUTOREJESTRACJI	409
Mieszanie i dopasowywanie podejść konfiguracji	415
12.3.Kiedy używać KONTENERA DI	416
Używanie bibliotek trzecich oznacza koszty i ryzyko	416
CZYSTE DI daje krótszy cykl feedbacku	419
Werdykt: kiedy używać KONTENERA DI	420
13 Kontener DI Autofac	425
13.1.Wprowadzenie do Autofac	426
Rozwiązywanie obiektów	427
Konfigurowanie ContainerBuilder	430
13.2.Zarządzanie cyklem życia	437
Konfigurowanie zakresów instancji	438
Zwalnianie komponentów	440

13.3.Rejestrowanie trudnych API	443
Konfigurowanie prymitywnych ZALEŻNOŚCI	443
Rejestrowanie obiektów za pomocą bloków kodu	445
13.4.Praca z licznymi komponentami	447
Wybieranie wśród wielu kandydatów	447
Podłączenie sekwencji	452
Podłączanie Dekoratorów	455
14. Kontener DI Simple Injector	458
Pisanie Kompozytów	463
14.1. Wprowadzenie do Simple Injector	464
Tworzenie obiektów	466
Konfigurowanie kontenera	468
14.2. Zarządzanie cyklem życia	476
Konfigurowanie STYLÓW ŻYCIA	477
Zwalnianie komponentów	478
Zakresy otoczenia	481
Diagnozowanie kontenera pod kątem częstych problemów cyklu życia	483
14.3. Rejestrowanie trudnych API	486
Konfigurowanie prymitywnych ZALEŻNOŚCI	486
Wyciąganie prymitywnych ZALEŻNOŚCI do Obiektów Parametrów	488
Rejestrowanie obiektów za pomocą bloków kodu	489
14.4. Praca z licznymi komponentami	490
Wybieranie spośród liczych kandydatów	491
Podłączanie sekwencji	494

Podłączanie Dekoratorów	497
Podłączanie Kompozytów	500
Sekwencje są strumieniami	502
15 Kontener DI Microsoft.Extensions.DependencyInjection	507
15.1.Wprowadzenie do	
Microsoft.Extensions.DependencyInjection	508
Rozwiązywanie obiektów	510
Konfigurowanie ServiceCollection	512
15.2.Zarządzanie cyklem życia	518
Konfigurowanie STYLÓW ŻYCIA	519
Zwalnianie komponentów	519
15.3.Rejestrowanie trudnych API	522
Konfigurowanie prymitywnych ZALEŻNOŚCI	522
Wyciąganie prymitywnych ZALEŻNOŚCI do Obiektów	
Parametru	523
Rejestrowanie obiektów za pomocą bloków kodu	524
15.4.Praca z licznymi komponentami	526
Wybieranie wśród liczych kandydatów	526
Podłączanie sekwencji	530
Podłączanie Dekoratorów	533
Podłączanie Kompozytów	537
słowniczek terminologii	545
literatura	549
Pozycje drukowane	549
Pozycje online	550
Inne źródła	551
indeks	553