

Spis treści

Wprowadzenie	11
---------------------------	-----------

CZĘŚĆ I FRAMEWORKI ZWINNE

Rozdział 1 Wprowadzenie do metodologii Scrum	21
Scrum kontra model kaskadowy	24
Role i obowiązki	26
Właściciel produktu	26
Scrum master	27
Zespół deweloperski	28
Artefakty	29
Tablica Scruma	29
Wykresy i wskaźniki	41
Rejestry	45
Sprint	47
Planowanie wydania	47
Planowanie sprintu	48
Codzienne spotkania Scruma	50
Demo sprintu	51
Retrospektywa sprintu	52
Kalendarz Scruma	54
Zwinność w prawdziwym świecie	55
Sztywność	55
Brak testowalności	56
Wskaźniki	57
Podsumowanie	59

Rozdział 2	Wprowadzenie do metodologii kanban	61
	Kanban — szybki początek	62
	Radiator informacyjny	62
	Ograniczanie pracy w trakcie	66
	Ochrona przed zmianą	66
	Definiowanie „ukończenia”	67
	Ceremonie sterowane zdarzeniami	68
	Klasy usług	69
	Umowy o gwarantowanym poziomie świadczenia usług	69
	Limity WIP dla klas usług	71
	Ludzie jako klasy usług	71
	Analiza	72
	Czas dostarczenia i czas cyklu	72
	Kumulatywne diagramy przepływu	73
	Podsumowanie	81

CZĘŚĆ II PODSTAWY TWORZENIA ADAPTYWNEGO KODU

Rozdział 3	Zależności i warstwy	85
	Zależności	86
	Prosty przykład	87
	Zależności platformowe	90
	Zależności zewnętrzne	92
	Modelowanie zależności za pomocą grafu skierowanego	93
	Zarządzanie zależnościami	97
	Implementacje i interfejsy	97
	Zapach instrukcji new	98
	Alternatywne sposoby tworzenia obiektów	101
	Rozwiązywanie zależności	103
	Zarządzanie zależnościami za pomocą projektów NuGet	112
	Warstwy	117
	Popularne wzorce warstwowania	118
	Zagadnienia przecinające	123
	Warstwowanie asymetryczne	124
	Podsumowanie	126
Rozdział 4	Interfejsy i wzorce projektowe	127
	Czym jest interfejs?	127
	Składnia	128
	Jawna implementacja	130
	Polimorfizm	134

Wzorce tworzenia adaptacyjnego kodu	135
Wzorzec Zerowy Obiekt	135
Wzorzec Adapter	140
Wzorzec Strategia	143
Dodatkowe funkcjonalności	145
Kacze typowanie	145
Domieszki	149
Płynne interfejsy	153
Podsumowanie	155
Rozdział 5 Testy	157
Testy jednostkowe	158
Przygotuj, wykonaj, zweryfikuj	158
Programowanie sterowane testami	162
Bardziej zaawansowane testy	166
Wzorce testów jednostkowych	180
Tworzenie elastycznych testów	180
Wzorzec Budowniczy testów jednostkowych	182
Wzorzec Budowniczy	182
Uwidaczanie przeznaczenia testu jednostkowego	183
Przed wszystkim testy	185
Co to jest TDD?	185
Wzorzec TDD	186
Wzorzec TFD	187
Inne testy	187
Piramida testów	188
Przeciwieństwa piramidy testów	189
Diagram testowy	190
Testy profilaktyczne i lecznicze	192
Jak zmniejszyć wskaźnik MTTR?	193
Podsumowanie	194
Rozdział 6 Refaktoryzacja	195
Wprowadzenie do refaktoryzacji	195
Zmiana istniejącego kodu	196
Nowy typ konta	204
Agresywna refaktoryzacja	208
Czerwone – zielone – refaktoryzacja... Przeprojektowanie	209
Adaptacja zastanego kodu	209
Technika złotego wzorca	210
Podsumowanie	216

CZĘŚĆ III SOLID-NY KOD

Rozdział 7	Zasada pojedynczej odpowiedzialności	219
	Opis problemu	219
	Refaktoryzacja poprawiająca czytelność kodu	222
	Refaktoryzacja zwiększająca abstrakcyjność kodu	226
	Zasada pojedynczej odpowiedzialności i wzorzec Dekorator	233
	Wzorzec Kompozyt	234
	Dekoratory predykatu	237
	Dekoratory warunkowe	240
	Leniwe dekoratory	241
	Dekoratory logujące	242
	Dekoratory profilujące	243
	Dekorowanie właściwości i zdarzeń	246
	Podsumowanie	247
Rozdział 8	Zasada „otwarty/zamknięty”	249
	Wprowadzenie do zasady „otwarty/zamknięty”	249
	Definicja Meyera	249
	Definicja Martina	250
	Usuwanie błędów	250
	„Świadomość” kodu klienckiego	251
	Punkty rozszerzeń	251
	Kod bez punktów rozszerzeń	251
	Metody wirtualne	252
	Metody abstrakcyjne	253
	Dziedziczenie interfejsu	254
	„Projektuj pod kątem dziedziczenia albo blokuj dziedziczenie”	254
	Chroniona zmienność	255
	Przewidywana zmienność	255
	Stabilny interfejs	255
	Właściwa adaptacyjność	256
	Przewidywana zmienność a spekulatywne uogólnienie	256
	Potrzebujesz aż tylu interfejsów?	257
	Podsumowanie	258
Rozdział 9	Zasada podstawienia Liskov	259
	Wprowadzenie do zasady podstawienia Liskov	259
	Oficjalna definicja zasady LSP	259
	Reguły tworzące zasadę LSP	260

Kontrakty	261
Warunki początkowe	262
Warunki końcowe	263
Inwarianty	264
Reguły kontraktowe w zasadzie LSP	266
Kontrakty kodu	272
Kowariancja i kontrawariancja	278
Definicje	278
Reguły systemu typów w zasadzie LSP	284
Podsumowanie	287
Rozdział 10 Segregacja interfejsów	289
Przykład segregacji interfejsów	289
Prosty interfejs CRUD	290
Zapamiętywanie danych	295
Dekorowanie wielu interfejsów	298
Tworzenie kodu klienckiego	300
Wiele implementacji, wiele instancji	301
Jedna implementacja, jedna instancja	303
Antywzorzec „interfejsowa mieszanka”	304
Dzielenie interfejsów	304
Wymagania kodu klienckiego	304
Wymagania architektury aplikacji	310
Interfejsy z pojedynczymi metodami	314
Podsumowanie	315
Rozdział 11 Odwracanie zależności	317
Planowanie zależności	317
Antywzorzec Świta	318
Wzorzec Schody	320
Przykład abstrakcyjnego projektu	321
Abstrakcje	322
Konkretne polecenia	322
Wyodrębnianie funkcjonalności	325
Ulepszony kod kliencki	329
Abstrakcje obsługujące zapytania	332
Dalsze abstrakcje	333
Podsumowanie	334

CZĘŚĆ IV STOSOWANIE ADAPTYWNYCH TECHNIK

Rozdział 12	Wstrzykiwanie zależności	337
	Skromne początki	337
	Aplikacja Lista zadań	340
	Tworzenie grafu obiektów	342
	Nie tylko proste wstrzykiwanie	358
	Antywzorzec Lokalizator Usług	358
	Wzorzec Nielegalne Wstrzykiwanie	361
	Korzeń kompozycji	363
	Konwencje zamiast konfiguracji	368
	Podsumowanie	371
Rozdział 13	Sprzężenie, spójność, współzależność	373
	Sprzężenie i spójność kodu	373
	Sprzężenie	373
	Spójność	374
	Współzależność	375
	Nazwa	376
	Typ	377
	Znaczenie	377
	Algorytm	378
	Pozycja	379
	Kolejność wykonania	379
	Czas	379
	Wartość	379
	Tożsamość	380
	Określanie stopnia współzależności	380
	Lokalność	380
	Nieoficjalna współzależność	380
	Współzależność statyczna i dynamiczna	381
	Podsumowanie	381

DODATKI

Dodatek	Adaptywne narzędzia	385
	Kontrola kodu źródłowego w systemie Git	385
	Tworzenie kopii repozytorium	387
	Przełączenie na inną gałąź	388
	Ciągła integracja	388
	Skorowidz	391