

Р 1572563

ROBERT C. MARTIN

РОБЕРТ МАРТИН

Чистий Agile

НАЗАД
ДО ОСНОВ

ВИДАВНИЦТВО

ФАБУЛА
#PRO

За майже 20 років, які минули від опублікування Маніфесту Agile, розуміння цієї методології стало суттєво відрізнятися від тих намірів, котрі було покладено в основу її творцями. Тож зараз саме час повернутися до оригінальної ідеї Agile: негроміздкого рішення маленького завдання, поставленого перед невеликими командами програмістів, які виконують дрібні завдання. А хто може допомогти в цьому краще за Дядька Боба — одного з авторів гнучкої розробки програмного забезпечення, відомого міжнародного консультанта та письменника Роберта С. Мартіна, чиї книжки «Чиста архітектура» та «Чистий код» стали настільними для багатьох програмістів.

Зміст

Передмова	11
Вступ	14
Слова подяки	17
1. Вступ до Agile	19
Історія Agile	21
Сноуберд	29
Після зустрічі в Сноуберді	32
Короткий огляд Agile	33
Правило хреста	33
Діаграми на стінах	34
Перше, що потрібно знати	37
Зібрання	37
Етап аналізу	38
Етап проектування	39
Етап реалізації	40
Гонка на виживання	41
Перебільшення?	41
Кращий спосіб	42
Нульова ітерація	43
Agile виробляє дані	44
Надія супроти управління	46
Уміння поводитися з правилом хреста	46
Послідовність реалізації функціоналу	50
Ось і огляду кінець	50
ЖИТТЄВИЙ ЦИКЛ	51
Висновки	54
2. Причини застосування Agile	55
Професіоналізм	56
Усюдище програмне забезпечення	57
Програмісти правлять світом	59
Катастрофа	61

Розумні очікування	62
Ми працюватимемо якісно! 62	
Постійна технічна готовність 63	
Стабільна продуктивність 65	
Недорога адаптивність 68	
Постійне вдосконалення 69	
Уміння бути безстрашним 69	
Перевірка контролю якості не повинна знайти нічогосінько 70	
Автоматизоване тестування 71	
Підстрахуємо одне одного 72	
Чесні оцінки 73	
Потрібно вміти казати «ні» 74	
Постійне наполегливе навчання 74	
Менторство 75	
Біль про права	75
Біль про права клієнта 75	
Біль про права розробника 76	
Клієнти 76	
Розробники 78	
Висновки.	80

3. Методи ділової взаємодії. 81

Планування	82
Триваріантна оцінка 83	
Історії та одиниці складності 84	
Історії банкоматів 85	
Історії 92	
Оцінювання історій 94	
Управління ітерацією 97	
Демонстрація 99	
Швидкість 100	
Невеликі релізи	101
Коротка історія управління вихідним кодом 102	
Стрічки 104	
Диски та системи контролю вихідного коду (SCCS) 105	
Subversion 105	
Git і тести 106	
Приймальне тестування.	107
Інструменти й методології 108	

Керована поведінкою розробка	109
Тим часом на практиці...	110
Цілісна команда	113
Спільне розташування	114
Висновки.	116
4. Методи командної роботи	117
Метафора	118
Предметно-орієнтоване проектування	120
Сталий темп, або рівномірна робота	121
Понаднормова робота	122
Марафон	123
Залученість і самовіддача	124
Сон	124
Колективна власність	125
Секретні файли	126
Безперервна інтеграція	128
А потім настав час безперервної збірки	129
Дисципліна під час безперервної збірки	130
Зустрічі навстоячки	131
Свині й кури?	132
Респект тобі!	132
Висновки.	132
5. Технічні методи	133
Керована тестами розробка	134
Подвійна бухгалтерія	135
Три правила керованої тестами розробки	136
Відлагодження	137
Документація	138
Задоволення	139
Повнота	139
Проектування	141
Сміливість	142
Рефакторинг	143
Червоний/Зелений/Рефакторинг	144
Значний рефакторинг	145
Просте проектування	146
Проектна вага коду	147

Парне програмування	148
Що таке парне програмування? 148	
Навіщо потрібне парне програмування? 149	
Парне програмування як огляд коду 150	
Які витрати на нас чекають? 150	
Лише два? 151	
Менеджери 151	
Висновки.	152

6. Упровадження Agile 153

Цінності Agile	154
Сміливість 154	
Комунікація 155	
Зворотний зв'язок 155	
Простота 156	
Усіляка всячина.	156
Трансформація	157
Виверти й відмовки 158	
Помста нового вожака 159	
Скиглії 159	
Мораль 160	
Фальсифікація 160	
Успіх у невеликих організаціях 161	
Індивідуальний успіх унаслідок переходу 161	
Створення Agile-організацій 162	
Коучинг.	163
Скрам-майстри 164	
Сертифікація	164
Справжня сертифікація 165	
Agile у масштабному застосуванні	165
Інструменти Agile	169
Інструменти розробки 169	
Що робить інструмент ефективним? 170	
Фізичні інструменти Agile 172	
Необхідність автоматизації 173	
Системи управління життєвим циклом розробки 175	
Коучинг — альтернативний погляд.	178
Шляхи, що ведуть до Agile 178	
Шлях до експерта з Agile 179	
Потреба в коучингу для Agile 180	

Перетворення звичайного коуча на <i>Agile</i> -коуча	181
Вихід за межі ICP-ACC	181
Коучингові інструменти	182
Професійних навичок коуча недостатньо	183
Коучинг у багатокомандному середовищі	183
<i>Agile</i> у масштабному застосуванні	184
Використання принципів <i>Agile</i> та коучингу для впровадження <i>Agile</i>	185
Посилення процесу впровадження <i>Agile</i>	186
Прямувати до великого, зосереджуватися на малому	188
Майбутнє <i>Agile</i> -коучингу	189
Висновки (і знову Боб)	190

7. Висока майстерність 191

Похмілля від <i>Agile</i>	193
Невідповідність очікувань і реальності	195
Віддаляємося одне від одного	197
Висока майстерність розробки	198
Ідеологія супроти методології	200
Чи є методи в майстерності розробки програмного забезпечення?	201
Сфокусуйтеся на цінностях, а не на методі	202
Обговорення методів	203
Вплив майстерності на особистість розробника	204
Вплив майстерності розробника на нашу галузь	205
Вплив майстерності розробника на компанії	206
Майстерність розробника й <i>Agile</i>	208
Висновки	208

8. Завершення 209

Післямова 211

Показчик 215

Про автора 223