

КЕНТ БЕК

ЭКСТРЕМАЛЬНОЕ ПРОГРАММИРОВАНИЕ

РАЗРАБОТКА ЧЕРЕЗ ТЕСТИРОВАНИЕ

TDD



Санкт-Петербург • Москва • Екатеринбург • Воронеж
Нижний Новгород • Ростов-на-Дону • Самара • Минск

2020

ББК 32.973.2-018
УДК 681.3.06
Б42

Бек К.

Б42 Экстремальное программирование: разработка через тестирование. — СПб.: Питер, 2020. — 224 с.: ил. — (Серия «Библиотека программиста»).

ISBN 978-5-4461-1439-9

Возвращение знаменитого бестселлера. Изящный, гибкий и понятный код, который легко модифицировать, который корректно работает и который не подкидывает своим создателям неприятных сюрпризов. Неужели подобное возможно? Чтобы достичь цели, попробуйте тестировать программу еще до того, как она написана. Именно такая парадоксальная идея положена в основу методики TDD (Test-Driven-Development — разработка, основанная на тестировании). Бессмыслица? Не спешите делать скороспелые выводы. Рассматривая применение TDD на примере разработки реального программного кода, автор демонстрирует простоту и мощь этой методики. В книге приведены два программных проекта, целиком и полностью реализованных с использованием TDD. За рассмотрением примеров следует обширный каталог приемов работы в стиле TDD, а также паттернов и рефакторингов, имеющих отношение к TDD. Книга будет полезна для любого программиста, желающего повысить производительность своей работы и получить удовольствие от программирования.

16+ (В соответствии с Федеральным законом от 29 декабря 2010 г. № 436-ФЗ.)

ББК 32.973.2-018
УДК 681.3.06

Права на издание получены по соглашению с Addison-Wesley Longman. Все права защищены. Никакая часть данной книги не может быть воспроизведена в какой бы то ни было форме без письменного разрешения владельцев авторских прав.

Информация, содержащаяся в данной книге, получена из источников, рассматриваемых издательством как надежные. Тем не менее, имея в виду возможные человеческие или технические ошибки, издательство не может гарантировать абсолютную точность и полноту приводимых сведений и не несет ответственности за возможные ошибки, связанные с использованием книги.

ISBN 978-0321146533 англ.
ISBN 978-5-4461-1439-9

© 2003 by Pearson Education, Inc.
© Перевод на русский язык ООО Издательство «Питер», 2020
© Издание на русском языке, оформление ООО Издательство «Питер», 2020
© Серия «Библиотека программиста», 2020

Оглавление

Предисловие	11
Благодарности	16
Введение.....	18
Часть I. На примере денег.....	21
Глава 1. Мультивалютные деньги	22
Глава 2. Вырождающиеся объекты	29
Глава 3. Равенство для всех.....	32
Глава 4. Данные должны быть закрытыми	36
Глава 5. Поговорим о франках.....	39
Глава 6. Равенство для всех, вторая серия.....	42
Глава 7. Яблоки и апельсины	47
Глава 8. Создание объектов.....	49
Глава 9. Потребность в валюте.....	53
Глава 10. Избавление от двух разных версий times().....	58
Глава 11. Корень всего зла	63
Глава 12. Сложение, наконец-то	66
Глава 13. Делаем реализацию реальной.....	71
Глава 14. Обмен валюты.....	76

Глава 15. Смещение валют.....	81
Глава 16. Абстракция, наконец-то!.....	85
Глава 17. Ретроспектива денежного примера	89
Что дальше?.....	89
Метафора.....	90
Использование JUnit	91
Метрики кода.....	92
Процесс	93
Качество тестов.....	94
Последний взгляд назад.....	95
Часть II. На примере xUnit.....	97
Глава 18. Первые шаги на пути к xUnit	98
Глава 19. Сервируем стол (метод setUp)	104
Глава 20. Убираем со стола (метод tearDown)	108
Глава 21. Учет и контроль	112
Глава 22. Обработка неудачного теста.....	115
Глава 23. Оформляем тесты в набор	118
Глава 24. Ретроспектива xUnit.....	123
Часть III. Шаблоны разработки через тестирование	125
Глава 25. Шаблоны разработки через тестирование	126
Тест	126
Изолированный тест (Isolated Test)	128
Список тестов (Test List)	129
Сначала тест (Test First)	131
Сначала оператор assert (Assert First).....	132
Тестовые данные (Test Data).....	134
Понятные данные (Evident Data).....	135
Глава 26. Шаблоны красной полосы.....	137
Тест одного шага (One Step Test).....	137
Начальный тест (Starter Test).....	138

Объясняющий тест (Explanation Test)	140
Тест для изучения (Learning Test).....	141
Еще один тест (Another Test)	142
Регрессионный тест (Regression Test).....	142
Перерыв (Break)	143
Начать сначала (Do over)	144
Дешевый стол, хорошие кресла (Cheap Desk, Nice Chair).....	145
Глава 27. Шаблоны тестирования.....	146
Дочерний тест (Child Test).....	146
Поддельный объект (Mock Object).....	147
Самошунтирование (Self Shunt).....	148
Строка-журнал (Log String)	150
Тестирование обработки ошибок (Crush Test Dummy)	151
Сломанный тест (Broken Test)	152
Чистый выпускаемый код (Clean Check-in).....	152
Глава 28. Шаблоны зеленой полосы	154
Подделка (Fake It)	154
Триангуляция (Triangulate).....	156
Очевидная реализация (Obvious Implementation).....	157
От одного ко многим (One to Many).....	158
Глава 29. Шаблоны xUnit	160
Проверка утверждений	160
Фикстура (Fixture).....	162
Внешняя фикстура (External Fixture)	164
Тестовый метод (Test Method).....	165
Тест исключения (Exception Test).....	167
Все тесты (All Tests)	168
Глава 30. Шаблоны проектирования.....	169
Команда (Command)	171
Объект-значение (Value Object)	172
Ноль-объект (Null Object)	174
Шаблонный метод (Template Method)	175
Встраиваемый объект (Pluggable Object).....	176
Встраиваемый переключатель (Pluggable Selector).....	178
Фабричный метод (Factory Method).....	180

Самозванец (Imposter)	181
Компоновщик (Composite)	182
Накапливающий параметр (Collecting Parameter)	184
Одиночка (Singleton)	185
Глава 31. Рефакторинг	186
Согласование различий (Reconcile Differences)	187
Изоляция изменений (Isolate Change)	188
Миграция данных (Migrate Data)	188
Выделение метода (Extract Method)	190
Встраивание метода (Inline Method)	191
Выделение интерфейса (Extract Interface)	192
Перемещение метода (Move Method)	193
Метод в объект (Method Object)	194
Добавление параметра (Add Parameter)	195
Параметр метода в параметр конструктора (Method Parameter to Constructor Parameter)	196
Глава 32. Развитие навыков TDD	197
Насколько большими должны быть шаги?	197
Что не подлежит тестированию?	198
Как определить качество тестов?	198
Как TDD способствует созданию инфраструктур?	199
Сколько должно быть тестов?	200
Когда следует удалять тесты?	202
Как язык программирования и среда разработки влияют на TDD?	203
Можно ли использовать TDD для разработки крупномасштабных систем?	203
Можно ли осуществлять разработку через тестирование на уровне приложения?	204
Как перейти к использованию TDD в середине работы над проектом?	205
Для кого предназначена методика TDD?	206
Зависит ли эффективность TDD от начальных условий?	207
Как методика TDD связана с шаблонами?	207
Почему TDD работает?	208
Что означает название?	210

Как методика TDD связана с практиками экстремального программирования?	210
Нерешенные проблемы TDD	212
Приложение I. Диаграммы взаимовлияния	213
Обратная связь	214
Контроль над системой	215
Приложение II. Фибоначчи	217
Послесловие	220