

Произошло действие или событие

Тогда (проверка)

Состояние системы изменилось или был произведен вывод

Кроме того, можно добавлять операторы, начинающиеся с **AND**, в любой из разделов ниже (Учитывая, Когда, Тогда).

Для примера требования шаги могут быть перечислены как:

Given

Book that has not been checked out

And

User who is registered on the system

When

User checks out a book

Then

Book is marked as checked out

Полный тест

Предыдущие шаги не включают каких-либо конкретных примеров данных, поэтому они добавляются для завершения теста:

Дано:

Книга, которая не была проверена

| Книги | |
|----------------|-----------|
| Заголовок | Проверено |
| Отличная книга | Нет |

Пользователь, который зарегистрирован в системе

| Пользователи | |
|--------------|-----|
| Имя | Сэм |

Когда:

Пользователь проверяет книгу

| Действие кассы | | | |
|----------------|-----|-----------|----------------|
| Пользователь | Сэм | Проверяет | Отличная книга |

Потом:

Книга отмечена как оформленная

| Книги | | |
|----------------|-----------|--------------|
| Заголовок | Проверено | Пользователь |
| Отличная книга | да | Сэм |

Тестовый экзамен

Изучение теста с конкретными данными обычно приводит к множеству вопросов. Для образца это могут быть:

- Что делать, если книга уже оплачена?
- Что делать, если книги не существует?
- Что делать, если пользователь не зарегистрирован в системе?
- Есть ли дата, когда книгу нужно сдать?
- Сколько книг может получить пользователь?

Эти вопросы помогают выявить недостающие или неоднозначные требования. К ожидаемому результату могут быть добавлены дополнительные детали, такие как срок выполнения. Другие приемочные тесты могут проверить, что такие условия, как попытка получить книгу, которая уже была извлечена, вызывают ожидаемую ошибку.

Еще один тестовый пример

Предположим, бизнес-клиент хотел иметь бизнес-правило, согласно которому пользователь может просматривать только одну книгу за раз. Следующий тест продемонстрирует, что:

Сценарий: убедитесь, что бизнес-правило оформления заказа применяется

Дано:

Книга, которая была проверена

| Книги | | |
|------------------------------|-----------|--------------|
| Заголовок | Проверено | Пользователь |
| Отличная книга | да | Сэм |
| Еще одна замечательная книга | Нет | |

| Пользователи |
|--------------|
| Имя |
| Сэм |

Когда:

Пользователь проверяет другую книгу

| Действие кассы | | | |
|----------------|-----|-----------|------------------------------|
| Пользователь | Сэм | Проверяет | Еще одна замечательная книга |

Потом:

Возникает ошибка

| Возникла ошибка |
|---------------------------------|
| Описание |
| Нарушение правил кассового дела |

Приемочные испытания проекта

Помимо приемочных испытаний требований, приемочные испытания могут использоваться для проекта в целом. Например, если это требование было частью проекта проверки библиотечных книг, приемочные испытания могли бы проводиться для всего проекта. Их часто называют целями SMART . Пример теста: «Когда новая библиотечная система находится в производстве, пользователи смогут регистрировать книги в три раза быстрее, чем сегодня».

Смотрите также

- [Concordion](#)
- [FitNesse](#)
- [Робот Фреймворк](#)
- [Датчик \(программное обеспечение\)](#)
- [Огурец \(программное обеспечение\)](#)

Рекомендации

Внешние ссылки

- [Пример фреймворков автоматизации \(http://acceptancetestdrivendevelopment.com\)](http://acceptancetestdrivendevelopment.com)

This page is based on the copyrighted Wikipedia article "Acceptance_test%E2%80%93driven_development" (Authors); it is used under the Creative Commons Attribution-ShareAlike 3.0 Unported License. You may redistribute it, verbatim or modified, providing that you comply with the terms of the CC-BY-SA.

abcdef.wiki is not affiliated with the Wikimedia Foundation

[Cookie-policy](#)

To contact us: mail to admin@qwerty.wiki

[Change privacy settings](#)