



Подходы к тестированию

Закрепление материалов лекции №4



Содержание

- Уровни ~~т~~естирования (типы тестов по покрытию)
- Приоритизация выполнения проверок
- Тестовые активности

Как вы поняли, что вы
протестировали
приложение **качественно**
?

(Ещё раз 😊) Тестовое ПОКРЫТИЕ —
это одна из метрик оценки
качества тестирования,
представляющая из себя
плотность покрытия тестами
требований либо исполняемого
кода

Тестовое покрытие определяется:

- Глубиной тестирования
- Шириной тестирования
- Целями тестирования
- Профессиональным уровнем тестирующего

Ширина тестирования

отражает количество модулей/функций приложения, которые были протестированы.

Глубина тестирования

отражает вид проверок, которые производились для модуля/функции (например, позитивность/негативность сценариев, тестирование в рамках/вне рамок тест сценариев и др.).

По покрытию (глубине)

- **Smoke** – проверка основной функциональности системы на валидных данных

Уровень качества: **Acceptable** / **Unacceptable**

- **MAT (Minimal Acceptance Test, Positive test)** – проверка каждой функции системы на валидных данных

Уровень качества: **High** / **Medium** / **Low**

- **AT (Acceptance Test)** – проверка всех возможных сценариев, включая негативные данные, защиту от дурака и пр.

Уровень качества: **High** / **Medium** / **Low**

По покрытию (глубине)



www.facebook.com/login/identify/

facebook

Электронный адрес или номер телефона

Пароль

[Вход](#)

[Забыли аккаунт?](#)

Найдите свой аккаунт

Введите ваш эл. адрес или номер телефона, чтобы найти свой аккаунт.

[Поиск](#) [Отмена](#)

По покрытию (ширине)

- **New Feature Test (NFT, AT of NF)** – тестирование новой функциональности, которая ранее не тестировалась
Обычно тестируется полным тестом (AT)



По покрытию (ширине)

- **Defect Validation** – проверка результата исправления дефекта, а также проверка того, что исправление не повлияло на ранее работавшую функциональность.

Может включать элементы регрессионного тестирования.



По покрытию (ширине)

- **Regression testing** – удостовериться, что изменения в коде (добавление новой функциональности или исправление дефектов) не повлияло на работу старой функциональности.

Регрессионное тестирование может быть проведено на уровне *Smoke*, *MAT* или *AT*.

Regression:
"when you fix one bug, you
introduce several newer bugs."



Тестовые активности:

- Smoke
- Регрессионное
- Валидация дефектов
- Тестирование новых функций

Как надо было
тестировать первое
задание?

1. Понять основную функциональность (через спецификацию, опыт, постановку себя на место конечного пользователя).
2. Провести Smoke (и понять можно ли тестировать дальше).
3. Подготовить себе тестовую документацию по приложению:
 - Разбить приложение на модули и функции;
 - Сгенерировать проверки для этих модулей и функций (на основе спеки, опыта, web-testing checklist) в соответствии с Smoke/MAT/AT.
4. Провести тесты по этой документации.

Вам предоставляется на тестирование веб-проект – интернет магазин по продаже деталей для автомобилей.

Из планов разработки известно, что на тестирование будут поставлены **3 сборки**:

- #1 включает в себя 50% функциональности
- #2 включает в себя оставшиеся 50% функциональности
- #3 является финальной (release candidate) и будет отдана заказчику

Вопрос:

- Какие тесты Вы проведете на каждой сборке?

Web Testing чеклист: [ссылка](#)
(помните, что можно добавить
проверки из чеклиста к своей
тестовой документации)

Закрепим

Сейчас вы разделитесь на две комнаты (две команды тестирования), я буду Вашим заказчиком, и вам необходимо:

- 1 – предложить мне тестовую документацию;
- 2 – предложить мне тесты для моего проекта.

Комната 1 – проект «Запор» - сайт для контроля заряда электромобилей

Комната 2 – проект «НеВсеДома» – социальная сеть для знакомства

**Спасибо за
внимание! Жду**

Ва

з

