

Введение в тестирование (что такое тестирование, основные принципы тестирования...)

Урок 5



План

1. Тестирование и его цели.
2. Основные принципы тестирования.
3. Цикл тестирования ПО.
4. Стратегия тестирования.




Тестирование и его цели.

Часто при разработке программного обеспечения приходится сталкиваться с одной из двух проблем. Либо качество разработанного продукта много ниже самых минимальных требований, либо затраты на тестирование превосходят все разумные пределы. Иногда, бывает и так, что обе проблемы существуют одновременно. И денег на тестирование истрчено много, а качества достичь не удалось.

Одной из причин такой ситуации является объективная сложность процесса тестирования ПО . Ведь под словом «Тестирование» может скрываться множество самых различных действий, направленных на решение разнообразных задач.

Например, запуск и исполнение программы с целью проверки отсутствия ошибок, и оценка производительности, и контроль наличия и полноты документации и даже качества принятых проектных решений. На этом уроке мы рассмотрим вопросы: «Что же такое тестирование? Каковы его цели и основные принципы? »



Тестирование - это проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом. Существует и другая формулировка.

Тестирование - это одна из техник контроля качества включающая в себя активности по планированию работ (Test Management), проектированию тестов (Test Designe), выполнения тестирования (Test execution) и анализа полученных результатов (Test analysis)



Цели тестирования (Testing goals).

1. Показать разработчику и клиенту, что программное обеспечение отвечает заявленным требованиям.
2. Найти ситуации, когда программное обеспечение ведет себя ошибочно, нежелательно или не соответствует спецификации.
3. Создать предпосылки для предупреждения возникновения ошибок в программном обеспечении.

Конечной целью любого процесса тестирования является обеспечение качества, с учётом всех или наиболее критичных для данного конкретного случая составляющих.



Основные принципы тестирования.

На этом уроке мы рассмотрим общеизвестные и основные принципы, о которых не следует забывать, когда тестируете какое либо приложение или новую функциональность. Из множества принципов, есть основные по мнению сообщества тестировщиков, которые и хотелось бы выделить:



Исчерпывающее тестирование – невозможно ни одним из тестировщиков.

Все понимают, что протестировать все возможные случаи и комбинации просто невозможно, конечно если это не тривиальный случай.

Просто невозможно проверить все возможные случаи, так как это заняло бы у нас очень много времени и в конце концов не стоило бы нам таких усилий. Если каждый тестировщик сядет и тщательно продумает все сценарии и если отдать эту фичу (Фича (англ. feature) — в жаргоне пользователей компьютеров, специфическая особенность, возможность или способность программы либо проекта к выполнению какой-либо функции.) для тестирования другому тестировщику, то он как правило, находит еще кучу возможных сценариев и кейсов, которые можно включить.

Поэтому в тестировании ПО принято проанализировать продукт или новую фичу, после чего сфокусировать усилия в тестировании на более рискованные и приоритетные случаи и участки нашего продукта.



Скопление багов

Если брать продукт и разбить его по модулям, то в процессе тестирования можно заметить что основная часть багов лежит в одном или нескольких модулях продукта, следовательно можно отметить эффект скопления багов.

Как правило, такое можно наблюдать в совершенно разных продуктах. Для того чтобы эффективно протестировать наш продукт, следует распределить свои усилия в тестировании по реальной плотности багов в модулях продукта, если же тестируем впервые, то пропорционально ожидаемой плотности.

Со временем тенденция с накоплением багов может изменяться от модуля к модулю. Это следует отслеживать и перераспределять усилия в дальнейшем тестировании.



Эффективность раннего тестирования.

Очень важно чтобы тестирование начиналось как можно раньше, и предугадывались возможные ошибки, которые может совершить разработчик.

Перед тем как разрабатывать и тестировать тот или иной продукт, следует выяснить всю специфику, возможные конфликты в спецификации, невозможность новой фичи взаимодействовать с другим модулем, убедиться что тестировщик, разработчик и product owner одинаково правильно понимают, как это будет реализовано.


Запомните чем раньше найдутся баги, тем дешевле их исправить.



Парадокс пестицида.

Если написав тест кейсы, прогонять их много раз, то в конечном счете эти кейсы не помогут нам найти новые баги. Поэтому существует практика в тестировании, когда пересматривают и модернизируют тест кейсы чтобы найти какие-то новые баги.

Тест кейсы могут усложняться, быть разносторонними так чтобы охватить все компоненты, модули нашего продукта, что в свою очередь поможет нам найти больше интересных и новых багов.




Тестирование зависит от нашего продукта.

Существует много программ, продуктов и к каждой из них следует подходить индивидуально в плане тестирования.

В каких-то больше усилий в тестировании нужно на безопасность, в каких-то на юзабилити.

Поэтому не стоит грести все продукты под одну гребенку и тестировать по какому-то одному шаблону.




Тестирование показывает наличие багов в продукте но ни как не их отсутствие.

Многие думают, что если новый функционал прошел стадию тестирования, значит все — багов уже нет. В этом и заключается ошибочное суждение.

Тестирование лишь снижает вероятность наличия багов в продукте.

В процессе тестирования много багов может быть пропущено и это не будет означать, что если продукт прошел тестирование, то сейчас этот продукт на 100% работает корректно.



Продукт хорошо протестирован и вроде бы нет очевидных багов, то это хороший продукт.

Иногда тестируя и выискивая функциональные баги, мы забываем посмотреть с другой стороны и спросить, а нужно ли это пользователю.

Если эта фича не соответствует ожиданиям пользователя и его потребностям, то какой бы качественный наш продукт не был — это уже не так важно.




Цикл тестирования ПО

Цикл тестирования ПО – это основное понятие внедрения тестирования как такового. Это понятие описывает все процессы, стадии, последовательности применения тестирования и доставки продукта конечному пользователю. Цикл тестирования применяется абсолютно для всех продуктов независимо от специфики.

Цикл тестирования ПО состоит из четырех этапов:

1. Изучение и анализ предмета тестирования.
2. Планирование тестирования.
3. Исполнение тестирования.
4. Анализ прошедшего тестирования и написание отчета о проведенном тестировании



Рассмотрим связь цикла тестирования с циклом разработки ПО

Изучение и анализ предмета тестирования

начинается перед утверждением спецификации (в завершении стадии «Разработка и дизайн продукта») и продолжаются на стадии «Кодирование».


Планирование тестирования.

Происходит на стадии «Кодирование»

Исполнение тестирования.

Происходит на стадии «Тестирование и отладка»


На любом из этапов может быть найден баг (как в ПО, так и в документации)



Изучение и анализ предмета тестирования.

Конечной целью этапа «Изучение и анализ предмета тестирования» является получение ответов на два вопроса:

- Какие функциональности предстоит протестировать?
- Как эти функциональности работают?



Основными источниками знания о функциональностях являются:

- документация (спецификации, ТЗ, блок-схемы и прочее)
- люди (информация постигается через межличностное общение)
- сам сайт или программа, которые мы изучаем посредством эксплоринга (исследования).

Эксплоринг- это изучение того, как работает программа с точки зрения пользователя.

В интернет компаниях эксплогинг применяется в случаях:

- когда написан код и отсутствует документация;
- для самообучения.

Например, в крупных интернет-компаниях вновь нанятые тестировщики в течение нескольких недель проходят тренинг, часть которого посвящена эксплорингу.



Планирование тестирования.

На этой стадии тестировщик принимает решение: «Как он будет тестировать продукт?». Эта стадия наиболее творческая и очень ответственная, потому что качество продукта напрямую зависит от принятых решений.

Исполнение тестирования

- это практический поиск багов в созданной программе или на сайте с использованием тест-кейсов и чек листов написанных ранее.



Стратегия тестирования.

Основополагающее понятие, как в тестировании, так и во всех жизненных областях. На этапе стратегии предопределяются вся дальнейшая политика, все дальнейшие действия, связанные с тестированием. Определяются стадии тестирования и типы тестов. Оформляется и разрабатывается тест план, определяется состав участников и их роли.

Стратегия определяет:

- какие инструментальные средства и методы будут применяться;
- критерии завершения итераций;
- процентную установку области охвата кода (то есть, что считать полностью протестированным приложением? 100% кода, или 95%);
- типы отчетов и способы автоматизированного их хранения;
- виды применяемых тестов: «черный ящик» или «белый ящик», либо их комбинации на различных стадиях