Принципы тестирования

В переводе с латинского При́нцип - это основа, начало, первоначало, и можно сказать, что принципы тестирования — это основы тестирования

1)-Исчерпывающее тестирование невозможно.

Насколько бы тщательным тестирование не было, нельзя учесть все возможные сценарии и предвидеть все возможные ошибки.

2️)- Тестирование демонстрирует наличие дефектов, а не их отсутствие.

Тестирование может выявить тот момент, что ошибки присутствуют, но не может доказать в полной мере, что дефектов нет. Даже если мы полностью проверили систему.

3)-Заблуждение об отсутствии ошибок.

Можно сколько угодно находить ошибки, и даже, казалось бы, не обнаруживая их больше, нет гарантии того, что ошибки найдены все и продукт полностью качественный и готовый.

Дефекты однозначно будут. Но в тестировании и нет такой задачи, чтобы выявить 100% багов, т.к. мы уже знаем, что это невозможно, исходя из первых трёх принципов. Главное здесь – найти наиболее критичные ошибки.

4)-Раннее тестирование сохраняет время и деньги

чем раньше выявляется та или иная проблема – тем меньше средств и трудозатрат потребуется для её устранения. Соответственно, если баг попадёт в «прод» или ещё хуже, если его найдёт пользователь – исправление такого дефекта обойдётся немалой кровью для всей команды. Помимо того, что удаление его будет стоить бо́льших денег, нежели на начальной стадии разработки, может получиться так, что этот дефект затронет другой функционал. И тогда проблемы начнут накапливаться как снежный ком. Сколько кода потребуется переписать разработчикам? Сколько времени уйдет на исправление и тестирование?

5)-Принцип скопления или кластеризация дефектов

Существует такое определение – наибо́льшее количество дефектов обычно содержится в небольшо́м количестве модулей. Простыми словами кластеризация – это группировка (на кластеры) множества объектов, схожих между собой по каким-либо параметрам. Представим полки и витрины в магазине – товары подразделены на хлебобулочные, молочные, мясные, напитки и др. Это и есть кластеризация.

Давайте проведём параллель с багами. Ошибки скапливаются в определённых местах, например, там, где код наиболее сложный или некорректно написан. Любой продукт состоит из модулей – кластеров в нашем случае. Если в каком-то модуле нашлось несколько багов, - это сигнал к тому, чтобы ещё внимательнее протестировать или даже перелопатить его с особой тщательностью на наличие скрытых дефектов.

6)-Тестирование зависит от контекста

Для разного софта будут применяться разные подходы к его тестированию. К примеру, способ тестирования мобильного приложения будет отличаться от того, которым тестируется коммерческий сайт.

По каким характеристикам различать контекст:

по типу продукта – web, desktop, мобильное приложение, сервис и др.;

по цели продукта – обеспечение безопасности, Game, продажа товаров и др.;

по проектной команде – специализация, количество человек, опыт и т.д.;

по доступным инструментам – что присутствует на проекте, для успешной реализации;

по срокам – как построен рабочий процесс, как часто выходят релизы, время между ними на подготовку;

по ожидаемому уровню качества – чем выше требования, тем тщательнее нужно тестировать.

7)-Парадокс пестицида

Если к какому-либо функционалу применять постоянно повторяющийся набор тестов – то эти проверки в скором времени будут неэффективны в нахождении новых дефектов.

Поэтому тест-кейсы должны постоянно обновляться и видоизменяться. Важно пользоваться такими рекомендациями:

-добавлять новые тесты;

-просматривать и изменять существующие;

-применять разные виды и техники тестирования;

-осуществлять тестирование новыми сотрудниками и др.

-В целом посмотреть на продукт под другим углом.