

Виды тестирования

по доступу кода

- метод серого ящика
 - когда мы можем протестировать часть кода (визуал+часть кода)
- метод чёрного ящика
 - Есть только визуал, доступа к коду нет
- метод белого ящика
 - позволяет протестировать продукт полностью, имея код + визуал

по запуску кода на исполнение

- Статическое
 - такой метод проводится, когда тестируем в основном требования. У нас есть некое ТЗ, есть документация, но продукт еще сам не разработан.
- Динамическое
 - У нас есть уже продукт и мы можем провести тестирование на рабочей системе

по уровню детализации приложения

- Модульное тестирование
 - логически выделенный или изолированный элемент. Тестирование проводится самими разработчики (Юнит тесты). У тестировщика должен быть доступ к коду (тестирование исходного кода, без визуальной части)
- Интеграционное тестирование
 - направлен на проверку взаимодействия сторонних интеграций (например, корректно ли чат-бот взаимодействует с сайтом)
- Системное тестирование
 - процесс тестирования системы, в котором проводится не только функциональное тестирование, но и оценка комплекса в системе, что наша система безопасна, устойчива и тд.
- Приёмочное тестирование
 - Метод при котором система проверяется на готовность к передаче заказчику (клиентам). Оценивается соответствие продукта бизнес-требованиям и требованиям пользователей

По степени автоматизации

- Ручное (мануальное)
 - это процесс поиска ошибок в программе без использования специальных ПО, силами человека
- Автоматизированное
 - выполняется с использованием специальных программных средств, которые, в свою очередь необходимы для выполнения набора тестовых примеров

по принципам работы с приложением

- Позитивное тестирование
 - проверяем, что продукт соответствует ожидаемому результату
- Негативное тестирование
 - когда продукт получает недопустимые или неожиданные входные данные.

По уровню функционального тестирования (по запуску тестов)

- Дымовое + санитарное тестирование
 - запуск продукта (если запустился - тестирование прошел), а потом проверить работу основных функций
- Критического пути
 - проверка функциональности, то есть проверка основных пользовательских функций
- Расширенное с тестирование
 - метод, при котором проводятся тесты, позволяющие оценить качество продукта на всех уровнях и в различных ситуациях. Используется для проверки работы приложения во всех возможных сценариях использования, учитывая все возможные варианты входных данных и условий работы программы

В зависимости от исполнителей

- Альфа тесты
 - закрытое тестирование, т.е. внутри команды
- Бета тестирование
 - подключаются к тестированию люди из вне(например пользователи)

В зависимости от целей тестирования

- Функциональное тестирование
 - модульное тестирование
 - логически выделенный или изолированный элемент. Тестирование проводится самими разработчики (Юнит тесты). У тестировщика должен быть доступ к коду (тестирование исходного кода, без визуальной части)
 - Дымовое + Санитарное
 - запуск продукта (если запустился - тестирование прошел), а потом проверить работу основных функций
 - Регресс
 - проверяем основные функции продукта после добавления новых фич
 - Альфа/Бета тестирование
 - тестирование с помощью собственных сил или с помощью людей из вне
 - Системное тестирование
 - проверка всей системы, то есть как и на ПО и как работает после внедрения новых функций
 - Интеграционное тестирование
 - направлен на проверку взаимодействия сторонних интеграций (например, корректно ли чат-бот взаимодействует с сайтом)
 - Критического пути
 - проверяем обычный функционал
 - E2E (энд ту энд)
 - тестирование на реальных конфигурациях, то есть когда тестирование проводится на якобы боевой версии продукта
- Нефункциональное тестирование
 - Производительности
 - проверка затрата памяти, нагрузки страницы и тд.
 - Нагрузочное тестирование
 - как система будет работать сверх лимита
 - Стресс
 - проверяем стабильность работы при максимальной нагрузке
 - Инсталляционное тестирование
 - проверка успешной установки, обновления и удаления нашего приложения, более предназначено для моб. тестирования
 - Интерфейса UI (пользовательский интерфейс)
 - Шрифты
 - Размер элементов
 - Цвета и тд.
 - Удобство пользования UX
 - Это проверка взаимодействия пользователя и сайта. Такое тестирование часто называют юзабилити тест-е
 - Локализация
 - проверка адаптации ПО для опред-ой аудитории (написание текста, даты, цвета и тд)
 - Надежности
 - цель проверка работоспособности приложения при длительном тестировании (довели до стрессового (или почти стрессового) напряжения и ждем как долго будет работать)
 - Безопасности
 - анализ рисков к защите целостности нашего подхода, атаке хакеров, вирусов, не санкционного доступа
 - Восстановление
 - когда система дает сбой и надо посмотреть как она восстанавливается
 - Кроссбраузерное тестирование
 - проверка того, что сайт отображается и работает везде одинаково

Проверка визуала контента