

Виды тестирования

По доступу кода

- Метод серого ящика
- Метод черного ящика
- Метод белого ящика

Процесс тестирования предоставляет особенность соединять метод черного и белого ящика, таким образом тестируем систему углубленно с доступом к исходному коду и интерфейс

Процесс тестирования проходит на требованиях и спецификациях не зная как работает система (нет доступа исходному коду) тестируем интерфейс.

Углубленное тестирование тесно связано с разработчиками, С известными системами с доступом к исходному коду, Со знанием особенностей тестируемой системы и ее дальнейшей работы

По запуску кода на исполнения

- Статическое
- Динамическое

Процесс тестирования требований, который проводим без запуска кода на обнаружение Багов (дефектов) в ПО. Составляем тест-кейсы

Тестирования на рабочей системе, фактическая эксплуатация тестируемого объекта как работает ее функционал соответствует ожиданиям или нет

По уровню детализации приложения

- Модульное тестирование
- Интеграционное тестирование
- Системное тестирование
- Приемочное тестирование

Полное тестирования с доступом к коду по методу белого ящика(тестируем код программы, без визуальной части)

Проверка взаимодействия функций

Оцениваем качество работы в комплексе с операционными системами (безопасность,устойчивость)

Проверяется требования пользователя, готовность системы последний этап перед выходом в release

По степени автоматизации

- Ручное (мануальное) тестирование
- Автоматизированное тестирование

Все действия тестирования проводятся в ручную

С использованием автотестов

По принципам работы с приложением

- Позитивное тестирование
- Негативное тестирование

Тестирование правильных действий и получаем ожидаемый результат

Тестирование по способу "Манки" то есть система получает недопустимые действия но при этом получаем ожидаемый результат в негативных ситуациях

По уровню функциональному тестированию

- Дымовое (смоки) и Санитарное (санити)тестирование
- Критического пути
- Расширенное тестирование

Запуск кода, проверка основных функций бизнеса (административных)

Проверяются основные пользовательские функции

Проверяем границы переполнения данных, ввод спец. символов не правильное нажатие кнопок ,возможности при не стандартном пользовании

В зависимости от исполнителей

- Альфа тесты
- Бета тесты

Закрытое тестирование, командное тестирование

Открытые тесты, с бета версией для обнаружения багов до полной готовности продукта выхода в release

В зависимости от целей тестирования

- Функциональное тестирование
- Нефункциональное тестирование

Модульное тестирование — Позволяет проверить правильность отдельных модулей исходного кода

Дымовое + Санитарное — Запуск кода проверяем основные функции работоспособности программы

Регресс — По добавлению изменений проверяем полностью работоспособность всей системы

Альфа/Бета тестирование — Проводим закрытые/открытые тесты

Системное тестирование — Качество работы с ОС

Интерграционное тестирование — Проверяем взаимодействие функций

Критические пути — Проверяем основные пользовательские функции

E2E — Создается копия сайта для тестирования в приближено к боевой версии

Производительность — Стабильность работы, (скорость загрузки)

Нагрузочное тестирование — Увеличение стандартной загружености

Стрес тест — Стабильность работы на максимальную загруженность

Инсталционное тестирование — Установка, настройка, обновление, удаление(тестирования предназначена для моб приложений)

Тестирование интерфейса UI — Проверка дизайна шрифт,размер ,цвет

Удобство использования UX — Насколько проект понятен удобен и легок в использовании

Локализация — Проверяем правильность перевода пользовательского интерфейса и документации всплывающих окон соответствие данных (валюты,даты)

Тестирование надежности — Проверяем работоспособность тестируемого продукта при длительной проверки, для понимания соответствует ли ожиданиям и требованиям

Тестирование безопасности — Анализ риска к защите приложения готовность к атаке хакера, несанкционированном доступе базы конфиденциальных данных, проникновения вирусов

Восстановление — Как восстанавливается после сбоя