

Виды тестирования

Функциональное

- Функциональные (функции и особенности нажатия кнопки , загрузки картинки) ( недостатки: логические ошибки, избыточное тестирование)
  - Требования (техническое задание, то что кнопки обрабатывают свои функции) что должно выполняться
  - Бизнес процессы (знания бизнес процессов, сценарии ежедневного использования) как должно выполняться
- Безопасности
  - Конфиденциальность (то что данные никуда не утекают, сокрытие определенной информации)
  - Целостность
    - Доверие (ожидается что какой либо ресурс будет изменен только существующим способом определенного круга людей)
    - Повреждение и восстановление (при повреждении определяется насколько важна функция восстановления)
  - Доступность (требования о том какие ресурсы будут доступны пользователю, залогиниться, выйти из аккауната)
- Взаимодействие (проверяет способность приложения взаимодействовать с одни или более компонентами)
  - совместимости (например с линуксом и тд)
  - интеграционное (карты, погода, банковские операции, биржа и тд)

Нефункциональное

- Производительности (как быстро система обрабатывает информацию под нагрузкой, граница приемлемой нагрузки)
  - Стресс тест (как быстро система вернется в нормальное состояние после стресс тестов)
  - Стабильности/надежности
- Установки (зачастую тестирование мобилок, геймдэв)
  - Система устанавливается
  - Обновляется
  - Удаляется
  - Настройка
  - Понижение версионности (даунгрейд)
- UX (тестирование удобства, если пользователь совершает больше 3х действий до конечного результата то это плохой Юзабилити Экспириенс)
  - Удобства использования
    - Производительность
    - Правильность( должно быть интуитивно понятно)
    - Активизация в памяти/recall
    - Эмоциональная реакция
    - Интуитивное поведение
  - Уровни проведения
    - Black box
    - Grey box
    - White box
- Отказ и восстановление (возможность восстановления после возникновения каких либо сбоев, отправили картинку и отключаем интернет)
- Конфигурация (кроссплатформенность, кросбраузерность)
- Локализация (как система воспринимает иностранные языки, арабская вязь и тд, тада на манер америки и тд.)
- UI (тестирование интерфейса, шрифты , цвета , Элементы, дизайн)

Связанные с изменениями

- Дымовое тестирование
  - Система стартует
  - Система выполняет основные функции
- Регресс (проверка того, что система работает так же после изменения багов и ничего не сломалось)
  - Регресс багов (попытка доказать что исправленная ошибка на самом деле не исправлена)
  - Регресс старых багов (
  - Регресс побочного эффекта
- UAT (проводятся перед релизом)
- Тестирование сборки (новая сборка соответствует требованиям)
- Санитарное тестирование (проверяет дефект в глубь)

По уровню тестирования

- Модульное(компонентное)/UNIT (ищем дефект в определенном модуле, обычно проверяет разработчик)
- Интеграционное(карты, погода, банковские операции, биржа и тд)
  - Снизу вверх (сначала тестируется низкокомодульные и по нарастающей)
  - Сверху вниз (сначала тестируется высокоуровневые модули и по уменьшению)
  - Большой взрыв (
- Системное тестирование
  - на базе требований
  - use case
- Приемочное тестирование\асептенс тесты (проверяет соответствие системы на требования, достиг ли продукт необходимого уровня качества) — UAT

По уровню доступа к коду

- Black box
    - Классы эквивалентности
    - Граничные значения
    - таблица принятия решений
    - Диаграмма состояний
    - Попарное тестирование
  - White BOX
  - Grey BOX
- Техники тест дизайна

По уровню выполнения тестов

- Ручное тестирование
- Автотестирование
  - Тестирование кода
  - Тестирование GUI/Интерфейс
  - Тестирование API