

Классификация видов тестирования

Виды тестирования

По доступности кода

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

Без доступа к коду. Тестирование основано на требованиях и спецификациях. Работа с внешним интерфейсом

Доступ к коду есть. Тестирование внутреннего устройства системы. Заведомо знаем входные данные и результат обработки

Комбинация белого и черного ящика. Тестирование проводим как и при черном ящике, тест-кейсы создаем, изучив код

По запуску кода на исполнение

- Статическое тестирование
- Динамическое тестирование

Преимущественно тестирование требований к ПО. Без запуска кода

Тестирование рабочей системы.

По уровню детализации приложения

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

Юнит тест, который проводит сам разработчик. Обязателен доступ к коду. Тестирование непосредственно кода

Проверка корректности работы модулей при совместной работе. Работа сторонней интеграции.

Проверка работа ПО в комплексе с ОС. Устойчивость, надежность, безопасность

Анализ соответствия требованиям пользователя/бизнес целям: возможность пользователя составить заявку, подписаться, указать номер телефона, осуществить запись/отмену и тд.

По степени автоматизации

- Ручное (мануальное) тестирование
- Автоматизированное тестирование

Прописывается автоматическая функция, которая выполняет тестирование. Использование специализированного ПО и языков тестирования.

По принципам работы с приложением

- Позитивное тестирование
- Негативное тестирование

Используем пользовательские сценарии с корректными значениями. (Пример: валидные данные при вводе пароля) Проверяем работоспособность заявленного функционала исходя из позитивного сценария

Проверяем как система реагирует на неожиданности. ПО не работает там, где не должен работать. Проверяем работоспособность заявленного функционала исходя из негативного сценария

По уровню функционального тестирования

- Дымовое +санитарное тестирование
- Тестирование критического пути
- Расширенное тестирование

Проверка запуска ПО и бизнес функций/ администратора

Тестирование функциональности, которую использует обычный пользователь ежедневно. Без функционала администратора

Направленно на тестирование всех заявленных требований функционала. Схоже с приемочным тестированием

В зависимости от исполнителя

- Альфа тест
- Бета тестирование

Проведение тестов командой производителей ПО

Тестирование ПО с привлечением реальных пользователей

В зависимости от целей тестирования

- Функциональное тестирование
- Нефункциональное тестирование

Модульное тестирование

Дымовое+санитарное

Регрессивное тестирование

Альфа/бета тестирование

Системное тестирование

Интеграционное тестирование

Тестирование критического пути

E2E или End-to-end

Проверяем как работает старый функционал ПО после внесения изменений/обновлений.

Проверяем функционирование системы после внесения изменений. Тестируют компоненты, прошедшие интеграционное тестирование

Функциональное тестирование на копии боевого сайта

Определение возможностей производительности ПО при различных условиях. Как влияют различные компоненты системы, условия использования пользователем и тд. Нет цели сломать ПО

Изучение работы системы при превышении норм нагрузки (при каких условиях "сломается")

Тестирование ПО на максимальных нагрузках

Проверяем работу ПО при выполнении операций скачать, установить, удалить

Тестирование UI

Тестирование UX

Локализация

Надежность

Безопасность

Восстановление

Кроссплатформенность/браузерность

Тестируем интерфейс ПО: шрифт, размер, цвет

Оцениваем удобство использования ПО

Интернационализация продукта, учет культурных особенностей и региона

Оцениваем способность работать при нагрузке

Оцениваем способность системы к самостоятельному восстановлению после сбоя

- Данные виды применимы для тестирования выбранного сайта. Не выбранные виды не подходят в связи с отсутствием доступа к полному коду; нет доступа к предыдущим версиям сайта и обновленным компонентам; информации о требованиях заказчика к ПО, бизнес целям и тд. Инсталляционное тестирование невозможно, тк объектом является сайт, а не приложение