

Виды тестирования

По доступу кода

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

* Тестирование «белого ящика» – с доступом к коду.
* Тестирование «черного ящика» – без доступа к коду продукта.
* Тестирование «серого ящика» – на основе ограниченного знания внутренней структуры ПО

По запуску кода на исполнение

- Статическое
- Динамическое

* Статическое тестирование – без запуска кода на исполнение. (Обнаруживает структурные ошибки, а также сюда относится проверка: документация, графических прототипов, параметров среды исполнения)
* Динамическое тестирование – с запуском кода, проверяется реальное поведение программы. (Обнаруживает: ошибки компиляции, ошибки периода выполнения, все ошибки нарушения логики)

По уровню детализации приложения

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

* Модульное (компонентное) тестирование – проверяются отдельные небольшие части приложения.
* Интеграционное тестирование – проверяется взаимодействие между несколькими частями приложения.
* Системное тестирование – приложение проверяется как единое целое.
* Приемочное тестирование – наиболее высокий уровень тестирования. Оно, также как и системное тестирование, необходимо для проверки работы программы в целом.

По степени автоматизации

- Ручное
- Автоматизированное

* Ручное тестирование – тест-кейсы выполняет человек.
* Автоматизированное тестирование – тест-кейсы частично или полностью выполняет специальное инструментальное средство.

По принципам работы с приложением

- Позитивное тестирование
- Негативное тестирование

* Позитивное тестирование – все действия с приложением выполняются строго по инструкции без никаких недопустимых действий, некорректных данных и т.д.
* Негативное тестирование – в работе с приложением выполняются (некорректные) операции и используются данные, потенциально приводящие к ошибкам.

По уровню функционального тестирования

- Дымовое + Санитарное
- Критического пути
- Расширенное тестирование

* Дымовое тестирование – проверка самой важной, самой ключевой функциональности, неработоспособность которой делает бессмысленной саму идею использования приложения.
* Тестирование критического пути – проверка функциональности, используемой типичными пользователями в типичной повседневной деятельности.
* Расширенное тестирование – проверка всей (остальной) функциональности, заявленной в требованиях.

В зависимости от исполнителей

- Альфа тесты
- Бета тестирование

Альфа- и Бета- тестирование используется, когда есть необходимость в получении обратной связи от пользователей. Поэтому именно они участвуют в таких проверках. Отличие альфа-тестирования от бета-тестирования заключается в том, что альфа-тестирование проводится внутри компании на потенциальных пользователях, а бета-тестирование проводится в ограниченном кругу конечных пользователей программы.

В зависимости от целей тестирования

Функциональное тестирование

- Модульное тестирование — Модульное (компонентное) тестирование – проверяются отдельные небольшие части приложения.
- Дымовое + Санитарное — дымовое- проверка работоспособности базовых функций, а санитарное- тщательная проверка "подозреваемой" функции
- Регресс — проверка не ухудшилась ли работа приложения после обновлений
- Альфа/Бета тестирование — альфа-тестирование проводится внутри компании на потенциальных пользователях, а бета-тестирование проводится в ограниченном кругу конечных пользователей программы
- Системное тестирование — полная проверка на ожидаемый результат
- Интеграционное тестирование — проверка взаимодействия двух и более компонентов
- Критического пути — проверка функциональности, используемой типичными пользователями в типичной повседневной деятельности
- E2E — проверка работы

Функциональное тестирование (functional testing) – направлено на проверку корректности работы функциональности приложения.

Нефункциональное тестирование

- Тестирование интерфейса UI
 - Шрифты
 - Размеры
 - Цвета
- Удобство использования UX — оценка удобства использования пользователями
- Локализация — оценка правильности версии программного продукта (языковой и культурный аспекты)
- Надежности — работа программы при длительной средней ожидаемой нагрузке
- Безопасности — проверка защиты от мошеннических действий
- Восстановление — проверка продукта на возможность сопротивления и успешного восстановления в случае возможных сбоев, возникших из-за ошибок ПО, оборудования или прерывания связи

Нефункциональное тестирование (non-functional testing) – тестирование атрибутов компонента или системы, не относящихся к функциональности.