

## **Разновидности тестирования: системное и регрессионное тестирование**

Системное тестирование (System Testing) — это этап тестирования ПО, на котором проверяется полностью интегрированная система как единое целое. Цель — убедиться, что приложение соответствует функциональным и нефункциональным требованиям в среде, максимально приближённой к реальной.

Ключевые характеристики:

- проводится после модульного и интеграционного тестирования;
- тестируется вся система в целом, включая взаимодействие между подсистемами, оборудованием, базами данных, сетями и сторонними сервисами;
- выполняется без знания внутренней структуры кода (метод «чёрного ящика»);
- может быть как ручным, так и автоматизированным.

Цели:

- проверить соответствие системы функциональным требованиям;
- оценить соответствие нефункциональным требованиям (производительность, безопасность, удобство использования и т.д.);
- выявить дефекты, связанные с неверным использованием ресурсов, непредвиденными комбинациями данных, несовместимостью с окружением;
- подтвердить корректность сквозных сценариев использования системы.

Виды системного тестирования:

- функциональное — проверка реализации заявленных функций;
- нагрузочное — оценка поведения системы под высокой нагрузкой;
- стресс-тестирование — проверка устойчивости при экстремальных нагрузках;
- тестирование безопасности — поиск уязвимостей и проверка механизмов защиты;

- юзабилити-тестирование (удобства использования) — оценка удобства интерфейса для конечных пользователей;
- тестирование совместимости — проверка работы на разных платформах, браузерах, устройствах;
- тестирование восстановления — проверка способности системы восстанавливаться после сбоев;
- конфигурационное тестирование — проверка работы при различных конфигурациях оборудования и ПО.

Пример: системное тестирование банковского приложения включает:

- вход в систему;
- выполнение транзакции;
- проверку конечного результата в выписке пользователя;
- тестирование работы приложения в разных браузерах и на мобильных устройствах;
- нагрузочное тестирование при одновременном доступе сотен пользователей.

---

## Регрессионное тестирование

Регрессионное тестирование (Regression Testing) — процесс повторного выполнения ранее проведённых тестов после внесения изменений в ПО (добавление новых функций, исправление ошибок, оптимизация), чтобы убедиться, что существующий функционал продолжает работать корректно.

Ключевая идея: любое изменение кода может случайно нарушить работу других частей системы. Регрессионные тесты помогают выявить такие побочные эффекты.

Когда выполняется:

- после исправления ошибок (багов);
- после добавления новой функциональности;
- после рефакторинга кода;
- после обновления зависимостей или среды выполнения;

- перед выпуском новой версии продукта.

Основные стратегии:

- полная регрессия —  
запуск всего набора тестов (надёжно, но ресурсоёмко);
- выборочная регрессия —  
выбор тестов, потенциально затронутых изменениями;
- приоритизация тестов —  
сначала выполняются тесты для критически важных функций;
- автоматизация —  
ключевой элемент эффективного регрессионного тестирования, особенно в CI/CD-пайплайнах.

Виды регрессионных тестов:

- юнит-тесты;
- интеграционные тесты;
- автоматизированные UI-тесты;
- тесты производительности;
- smoke-тесты (дымовые тесты) для быстрой проверки стабильности сборки.

Этапы проведения:

1. Идентификация изменений в коде.
2. Выбор стратегии регрессии (полная/выборочная).
3. Отбор соответствующих тест-кейсов.
4. Выполнение тестов (вручную или автоматически).
5. Анализ результатов и фиксация дефектов.
6. Повторение цикла при необходимости.

Пример: в интернет-магазине после исправления бага в корзине покупок выполняется регрессионный набор тестов:

- проверка добавления товара в корзину;
- расчёт итоговой суммы с учётом скидок;

- оформление заказа;
- проверка работы поиска товаров;
- авторизация пользователя;
- просмотр истории заказов.

Это гарантирует, что исправление бага в корзине не нарушило другие функции сайта.

---

## Сравнение системного и регрессионного тестирования

| Параметр         | Системное тестирование                                     |
|------------------|------------------------------------------------------------|
| Цель             | Проверить систему в целом на соответствие требованиям      |
| Когда проводится | После интеграции всех компонентов                          |
| Объём            | Сквозные сценарии по всей системе                          |
| Фокус            | Соответствие функциональным и нефункциональным требованиям |
| Тип тестирования | Чаще «чёрный ящик»                                         |
| Автоматизация    | Частично автоматизируется                                  |