

ВВЕДЕНИЕ В ТЕСТИРОВАНИЕ ПО. МЕТОДЫ И ВИДЫ ТЕСТИРОВАНИЯ



АНАСТАСИЯ ШАРИКОВА



АНАСТАСИЯ ШАРИКОВА

QA Lead в Bookmate



shharikova@gmail.com




[@shharikova](https://www.instagram.com/shharikova)



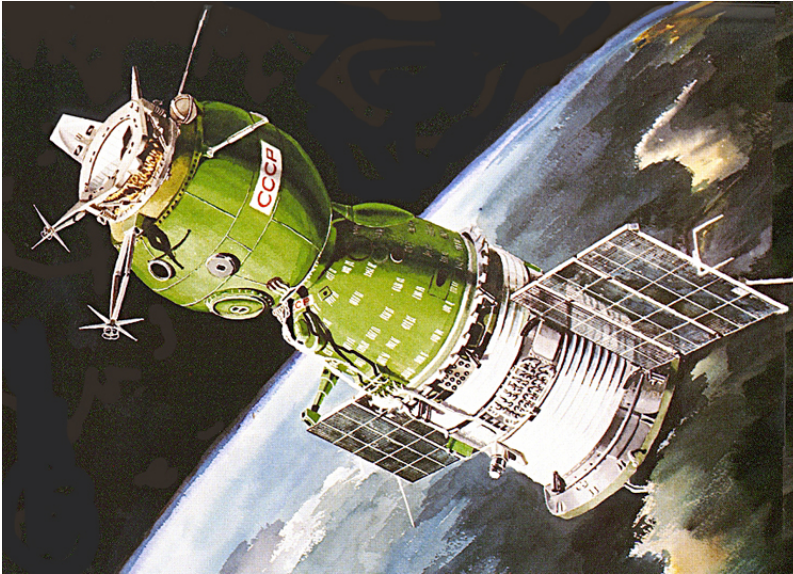
ПЛАН ЗАНЯТИЯ

1. Основные понятия. Что такое тестирование. История тестирования.
2. Цели тестирования
3. Цикл тестирования ПО
4. Уровни и типы тестирования



**ОСНОВНЫЕ ПОНЯТИЯ.
ЧТО ТАКОЕ ТЕСТИРОВАНИЕ.
ИСТОРИЯ ТЕСТИРОВАНИЯ.**

СОЮЗ-1



23 апреля 1967 года при досрочном возвращении корабля «Союз-1» на Землю отказала парашютная система. Корабль разбился недалеко от города Орска Оренбургской области. Космонавт Владимир Комаров погиб.



СОЮЗ-1

Трагическая авария корабля Союз-1 — печальный пример, что бывает, если в спешке перед финальным запуском проекта забыть про негативные сценарии тестирования.

Поэтому теория тестирования — очень важная часть знаний для начинающего профессионального специалиста по качеству.



ИСТОРИЯ ТЕСТИРОВАНИЯ

История контроля качества начинается задолго до эпохи активного развития разработки ПО — например, в виде ГОСТов или отделов ОТК на заводах.

Если говорить про разработку ПО и место тестирования в ней, то в 60-70-е годы тестирование было отделено от процесса разработки, в 80-е оно стало проводиться на протяжении всего жизненного цикла разработки ПО.



КТО ТАКОЙ ТЕСТИРОВЩИК СЕГОДНЯ?

В 90-х годах произошёл переход от тестирования как такового к более всеобъемлющему процессу, который называется «обеспечение качества (quality assurance)», охватывает весь цикл разработки ПО и затрагивает процессы планирования, проектирования, создания и выполнения тест-кейсов, поддержку имеющихся тест-кейсов и тестовых окружений.



КТО ТАКОЙ ТЕСТИРОВЩИК СЕГОДНЯ?

Простыми словами — тестировщик сегодня это человек, который помогает делать продукт лучше с каждым днем, используя современные технологии и методики :)



ТЕСТИРОВАНИЕ ЭТО...

Тестирование — проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом.

QA, QC & TESTING



Место, которое занимают QA, QC & Testing в процессе разработки



QA, QC & TESTING

- **Обеспечение качества (QA)** отвечает за весь процесс разработки, поэтому должно быть интегрировано во все этапы разработки: от описания проекта до тестирования, релиза и даже пост-релизного обслуживания.

Специалисты QA создают и реализуют различные тактики для повышения качества на всех стадиях производства: подготовка и установление стандартов, анализ качества, выбор инструментов, предотвращение появления ошибок и постоянное усовершенствование процесса.



QA, QC & TESTING

- **Задача Контроля качества (QC)** — гарантировать соответствие требованиям (поиск ошибок и их устранение).

QC ориентирован на проверку продукта, включает в себя многие процессы, такие как анализ кода, технические обзоры, анализ дизайна, тестирование и пр.

- **Тестирование** — это проверка результатов работы на соответствие требованиям.



ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

- Дефект/Баг/Дефект Репорт (Bug Report);
- Тестовый сценарий (Test Case);
- Тестовые данные (Test Data);
- План Тестирования (Test Plan);
- Тест дизайн (Test Design);
- Тестовое Покрытие (Test Coverage).



ДЕФЕКТ/БАГ/ДЕФЕКТ РЕПОРТ

Дефект / Баг / Дефект Репорт (Bug Report) — это документ, описывающий ситуацию или последовательность действий, которая привела к некорректной работе объекта тестирования, с указанием причин, фактического и ожидаемого результата.



ТЕСТОВЫЙ СЦЕНАРИЙ

Тестовый сценарий (Test Case) — это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.

ТЕСТОВЫЙ СЦЕНАРИЙ

Пример тестового сценария, в данном случае — на создание страницы:

Zephyr Test Management / TM-4

Create Page

Edit

Comment

Start Progress

Resolve Issue

Close Issue

Admin

Clone

More Actions

Execute

Details

Type: Test

Priority: Medium

Labels: None

Status: OPEN (View Workflow)

Resolution: Unresolved

Description

Click to add description

Test Details

	Test Step	Test Data	Expected Result
1	Click on "Ad new Page"	select "empty page"	A pop-up appears on which you can select the templates.
2	Click on "Add Page"		An empty page gets created and you see it in edit-mode.
3	Enter a page title in the given field.	Test Title - 1	You can enter a page title and you get the information that this title is not used up to now.
4	Enter text to the page content	Lorem ipsum dolor sit amet,	You can enter the text to the content section.
5	Click "Save"		The page gets saved and you see the page in view-mode.

Add



ТЕСТОВЫЕ ДАННЫЕ

Тестовые данные — набор данных, необходимых для выполнения тестового сценария.

Ими могут быть, например, логин и пароль или файл для загрузки в программу, на котором загрузчик перестает работать.



ПЛАН ТЕСТИРОВАНИЯ

План Тестирования (Test Plan) — это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения.

ПЛАН ТЕСТИРОВАНИЯ

Пример составляющих тест-плана на основе стандарта ISO/IEC/IEEE 29119:

- Контрольные даты;
- Общая информация: о проекте, цели тестирования, принципы тестирования;
- Что мы будем тестировать;
- Что мы не будем тестировать;
- Критерии входа;
- Критерии выхода;
- Риски;
- Команда, работающая над проектом;
- Тестовая среда;
- Временная шкала по итерации.



ТЕСТ ДИЗАЙН

Тест дизайн (Test Design) — это этап процесса тестирования ПО, на котором проектируются и создаются тестовые случаи (тест кейсы), в соответствии с определёнными ранее критериями качества и целями тестирования.



ТЕСТ ДИЗАЙН

- в идеале, за этот процесс отвечаю тест-аналитик и тест-дизайнер, но в реалиях небольших команд это могут делать и рядовые QA специалисты.
- первым этапом проводится анализ имеющихся проектных артефактов: документация (спецификации, требования, планы), модели, исполняемый код и т.д.
- далее происходит написание спецификации по тест дизайну (Test Design Specification).
- и, наконец, проектирование и создание тестовых случаев (Test Cases).



ТЕСТОВОЕ ПОКРЫТИЕ

Тестовое Покрытие (Test Coverage) — это одна из метрик оценки качества тестирования, представляющая из себя плотность покрытия тестами требований либо исполняемого кода.

ТЕСТОВОЕ ПОКРЫТИЕ

- Покрытие требований (Requirements Coverage) — оценка покрытия тестами функциональных и нефункциональных требований к продукту путем построения матриц трассировки (traceability matrix).
- Покрытие кода (Code Coverage) — оценка покрытия исполняемого кода тестами, путем отслеживания непроверенных в процессе тестирования частей программного обеспечения.
- Тестовое покрытие на базе анализа потока управления — оценка покрытия, основанная на определении путей выполнения кода программного модуля и создания выполняемых тест кейсов для покрытия этих путей.



ЦЕЛИ ТЕСТИРОВАНИЯ



ЦЕЛИ ТЕСТИРОВАНИЯ

Цель тестирования — минимизация количества существующих дефектов в конечном продукте. Грамотно организованное тестирование дает гарантию того, что:

- система удовлетворяет требованиям;
- система ведет себя в соответствии с требованиями во всех предусмотренных ситуациях.

Задача тестирования — определение условий, при которых проявляются дефекты системы.



А ЕЩЕ?

А еще тестирование помогает экономить ресурсы и дает нашему продукту конкурентные преимущества.



ЦИКЛ ТЕСТИРОВАНИЯ ПО



ЦИКЛ ТЕСТИРОВАНИЯ ПО

Основные процессы тестирования ПО:

- Планирование;
- Мониторинг и контроль;
- Проектирование и анализ;
- Реализация и выполнение тестов;
- Создание отчетности;
- Завершение и подведение итогов.

Попробуем рассмотреть их на примере релиза новой версии приложения.



ПЛАНИРОВАНИЕ

- Определение целей тестирования;
- Управление тестированием.

Цель — проверить, что существующий функционал функционирует как и ранее, ошибки исправлены, а новые фичи внедрены верно.

Решаем, сколько ресурсов у нас есть на тестирование.



ПРОЕКТИРОВАНИЕ И АНАЛИЗ

- Рецензирование основы тестирования (целостность, анализ рисков, архитектура, дизайн);
- Расстановка приоритетов;
- Подготовка окружения.

*Продумываем, в какой функционал наше «слабое» место, решаем, на что бросить основные силы, подготавливаем тестовые данные, такие как логины и пароли — задача в том, чтобы понять **что** тестировать и **как**.*



РЕАЛИЗАЦИЯ И ВЫПОЛНЕНИЕ ТЕСТОВ

- Написание тестовых сценариев;
- Проведение тестирования.

Пишем сценарии для нового функционала, проходим тесткейсы.



СОЗДАНИЕ ОТЧЕТНОСТИ

- Анализ проведенной работы;
- Выводы о том, как использовать новые знания и как использовать их в будущей работе.

Делаем выводы: например, как лучше тестировать новый функционал.



ДОПОЛНИТЕЛЬНЫЕ ЭТАПЫ

- Тестирование сопровождения;
- Работа с отзывами от пользователей;
- И многие другие...

Например, изучаем отзывы бета-пользователей.



УРОВНИ И ВИДЫ ТЕСТИРОВАНИЯ



КЛАССИФИКАЦИЯ ТЕСТИРОВАНИЯ

Существует огромное количество способов классифицировать деятельность в сфере тестирования, что нужно для того, чтобы максимально эффективно совмещать различные виды работ.

Сегодня же мы начнем с классификации по уровням тестирования и по задачам и целям.



УРОВНИ ТЕСТИРОВАНИЯ

Уровень тестирования определяет то, над чем производятся тесты: над отдельным модулем, группой модулей или системой, в целом. Проведение тестирования на всех уровнях системы — это залог успешной реализации и сдачи проекта.

- Компонентном или модульном (Component testing);
- Интеграционном (Integration testing);
- Системном (System testing);
- Приемочном (Acceptance testing).



КОМПОНЕНТНОЕ ИЛИ МОДУЛЬНОЕ

Компонентное (модульное) тестирование проверяет функциональность и ищет дефекты в частях приложения, которые доступны и могут быть протестированы по-отдельности (модули программ, объекты, классы, функции и т.д.).



ИНТЕГРАЦИОННОЕ

В данном случае проверяется взаимодействие между компонентами системы после проведения компонентного тестирования.



СИСТЕМНОЕ

Полностью реализованный программный продукт подвергается системному тестированию. На данном этапе тестировщика интересует не корректность реализации отдельных процедур и методов, а вся программа в целом, как ее видит конечный пользователь.

Основой для тестов служат общие требования к программе, включая не только корректность реализации функций, но и производительность, время отклика, устойчивость к сбоям, атакам, ошибкам пользователя и т.д.




ПРИЕМОЧНОЕ

Формальный процесс тестирования, который проверяет соответствие системы требованиям и проводится с целью:

- определения удовлетворяет ли система приемочным критериям;
- вынесения решения заказчиком или другим уполномоченным лицом принимается приложение или нет.

Проводится, когда продукт стал соответствовать изначальным требованиям ТЗ.



ВИДЫ ТЕСТИРОВАНИЯ НА ОСНОВЕ ЦЕЛЕЙ И ЗАДАЧ



ФУНКЦИОНАЛЬНОЕ ТЕСТИРОВАНИЕ

Функциональное тестирование — рассматривает заранее указанное поведение и основывается на анализе спецификаций функциональности компонента или системы в целом. А проще говоря, тестирование функциональности на соответствие требованиям.

Функциональные тесты могут быть представлены на всех уровнях тестирования, рассмотренных выше.



ТЕСТИРОВАНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

Тестирование пользовательского интерфейса (GUI Testing) — функциональная проверка интерфейса на соответствие требованиям — размер, шрифт, цвет, верстка.



НЕФУНКЦИОНАЛЬНЫЕ ТЕСТЫ

Нефункциональное тестирование системы выполняется для оценки таких характеристик системы и программного обеспечения, как удобство использования, производительность или безопасность.



ТЕСТИРОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ

Виды тестирования производительности:

- нагрузочное тестирование (Performance and Load Testing);
- объемное тестирование (Volume Testing);
- стрессовое тестирование (Stress Testing);
- масштабируемости (Scalability);
- конкурентное (Concurrency testing).

ТЕСТИРОВАНИЕ УДОБСТВА ПОЛЬЗОВАНИЯ (USABILITY TESTING)

Тестирование удобства пользования — это метод тестирования, направленный на установление степени удобства использования, обучаемости, понятности и привлекательности для пользователей разрабатываемого продукта в контексте заданных условий.





ТЕСТИРОВАНИЕ БЕЗОПАСНОСТИ (SAFETY TESTING)

Тестирование программного продукта с целью определить его безопасность.

Помимо этого:

- анализ рисков;
- атаки хакеров;
- вирусов;
- несанкционированного доступа к конфиденциальным данным.



СВЯЗАННЫЕ С ИЗМЕНЕНИЯМИ ВИДЫ ТЕСТИРОВАНИЯ

После проведения необходимых изменений, таких как исправление бага/дефекта, программное обеспечение должно быть перетестировано для подтверждения того факта, что проблема была действительно решена.

О таких видах — расскажу далее:



ДЫМОВОЕ ТЕСТИРОВАНИЕ (SMOKE TESTING)

Дымовое тестирование рассматривается как короткий цикл тестов, выполняемый для подтверждения того, что после сборки кода (нового или исправленного) устанавливаемое приложение стартует и выполняет основные функции.



РЕГРЕССИОННОЕ ТЕСТИРОВАНИЕ (REGRESSION TESTING)

Регрессионное тестирование — это вид тестирования, направленный на проверку изменений, сделанных в приложении или окружающей среде (починка дефекта, слияние кода, миграция на другую операционную систему, базу данных) для подтверждения того факта, что существующая ранее функциональность работает как и прежде.



ТЕСТИРОВАНИЕ СБОРКИ (BUILD VERIFICATION TEST)

Тестирование, направленное на определение соответствия выпущенной версии, критериям качества для начала тестирования. По своим целям является аналогом Дымового тестирования, направленного на приемку новой версии в дальнейшее тестирование или эксплуатацию.



САНИТАРНОЕ ТЕСТИРОВАНИЕ ИЛИ ПРОВЕРКА СОГЛАСОВАННОСТИ/ ИСПРАВНОСТИ (SANITY TESTING)

Санитарное тестирование — это узконаправленное тестирование, достаточное для доказательства того, что конкретная функция работает согласно заявленным в спецификации требованиям.



И ЕЩЕ НЕКОТОРЫЕ ЦЕЛИ



ТЕСТИРОВАНИЕ УСТАНОВКИ (INSTALLATION TESTING)

Тестирование установки направлено на проверку успешной инсталляции и настройки, а также обновления или удаления программного обеспечения.

Например, сборка мобильного приложения может прекрасно работать сама, но если установить ее поверх предыдущей версии из AppStore, она даже не будет запускаться из-за внутреннего конфликта.



ТЕСТИРОВАНИЕ НА ОТКАЗ И ВОССТАНОВЛЕНИЕ (FAILOVER AND RECOVERY TESTING)

Этот вид тестирования проверяет тестируемый продукт с точки зрения способности противостоять и успешно восстанавливаться после возможных сбоев, возникших в связи с ошибками программного обеспечения, отказами оборудования или проблемами связи (например, отказ сети).



КОНФИГУРАЦИОННОЕ ТЕСТИРОВАНИЕ (CONFIGURATION TESTING)

Специальный вид тестирования, направленный на проверку работы программного обеспечения при различных конфигурациях системы (заявленных платформах, поддерживаемых драйверах, при различных конфигурациях компьютеров и т.д.).



КРАТКИЕ ВЫВОДЫ

Сегодня мы начали узнавать основные понятия, важные для тестировщика и немного приблизились к пониманию того, чем же на самом деле занимается специалист по качеству.

Сфера тестирования одновременно сложна и интересна тем, что в ней нет стандартизированного сценария развития как специалиста, так и тем, что в каждой компании и даже команде вам придется использовать совершенно разные методики, перечисленные выше.



КРАТКИЕ ВЫВОДЫ

Одна из наших целей, научиться находить баланс между тем, сколько ресурсов есть для разнообразных задач, сколько нужно (и можно) потратить и какими методами это сделать.

Тестирование в реальном мире, зачастую — оно не про идеальный софт, а про понимание «уровня совершенства», который применением к каждой отдельной ситуации.

Например, что если у нас есть час на все тесты, лучше провести смок тестирование, а когда неделя — мы можем спокойно провести полноценный регресс.

P.S.

Конечно, сегодня мы рассмотрели только основные цели, понятия и методы, на самом деле их намного больше, но с помощью такого набора можно составить представление о работе отдела тестирования в целом, что, по-моему, самое главное в первой лекции.

ДОМАШНЕЕ ЗАДАНИЕ

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задаем в чате в Slack.
- Задания можно сдавать по частям.
- Также пройдите тестирование в личном кабинете Нетологии.
- Зачет по домашней работе проставляется после того, как приняты **все задания**



Спасибо за внимание!
Время задавать вопросы.

АНАСТАСИЯ ШАРИКОВА



shharikova@gmail.com



[@shharikova](https://www.instagram.com/shharikova)