Тестирование - Процесс проверки работоспособности по (сайта, программы) на предмет соответствия требованиям предъявляемым данному программному обеспечению.

**или**

Тестирование программного обеспечения — процесс анализа программного средства и сопутствующей документации с целью выявления дефектов и повышения качества продукта.

**Testing** - процесс помогающий определить корректность, полноту и качество разрабатываемого ПО. В тестировании используются различные методы, инструменты, техники для обнаружения и исправления дефектов в продукте. Проще говоря это сам процесс когда вы кликаете, проверяете что всё работает как задумано. Мы занимаемся конкретно тестированием. Можно сказать это самый нижний уровень где всё тестирование проходит по факту.

**QC**- quality control - контроль качества программного продукта. Выявление и исправление дефектов в готовом продукте. Т.е. речь идёт об поиске багов в исключительно готовом продукте. QC - это technical activity тогда как qa это menejment activity (управленческая деятельность).

-QC - detecting defects.

-Product oriented.

-for testing phase in SDLC

**Все тестеры (manual and automation) относятся к QC а не к QA** Работники qa определяют процесс разработки ПО, а qc следуют этому процессу.

**QA** - quality assurance - обеспечение качества **процесса** разработки ПО.

Включает в себя планирование, проектирование, реализацию стратегий и процессов разработки ПО в целом (т.е. как ПО будет разрабатываться, какие стандарты будем использовать, как тестироваться и т.д.).

Другими словами Обеспечивает процесс создания ПО в определённом векторе где просто разработать ПО недостаточно, надо сделать это следуя определённым правилам.

QA - preventing defects.

QA - Process oriented (are we following this or that…).

QA - for the entire life cycle of a SDLC.

Например планирование обучения новой технологии для сотрудников, благодаря чему повысится качество разрабатываемого продукта.

**Quality assurance - part of quality management focused on providing confidence that quality requirements will be fulfilled. [ISTQB Glossary]**

Пример: Внедрение новой процедуры по одобрению кода разработчиков

Основные понятия:

**Ошибка** - (Error) действие человека приводящие к неверному результату (например: программист допустил опечатку в коде), в следствие чего это привело к неверному поведению программы (дефект or Bug).

**Дефект** (defect or Bug) - это несоответствие продукта или компонента продукта его спецификациям или ожиданиям. Actual result and expected result. Может быть обнаружен как во время тестирования продукта так и после выпуска продукта.

или

Дефект — расхождение ожидаемого и фактического результата

**Баг репорт** - в нём структурировано, детально описывается как именно воспроизводится баг и в чём собственно баг выражается.

**Отказ** (Failure) - когда дефект обнаружен пользователем.

**Positive test** - проверка на правильных данных (пользователь вводит всё корректно)

**Negative test** - проверка на неправильных данных, реакция на некорректные действия. Ввод каких то не корректных, нелогичных данных при этом система должна корректно отрабатывать.

**Test case** - последовательность шагов которые нужно выполнить для того чтобы проверить работоспособность нашего продукта (приложения, сайта, программы). Т.е. это в какой то мере пошаговая инструкция что нужно сделать и какой результат мы ожидаем.

**Build -** некая промежуточная версия продукта

**Pass Criteria** - установленное правило, позволяющее определить удачно или неудачно прошел тест.

**Фича** - единица функциональности продукта. Используется в контексте наработки нового функционала (мы хотим добавить новую фичу в наш продукт) или в контексте обращения к той функциональности которая уже существует (фича авторизации работает некорректно)

**Спецификация** (спека) - требования оформленные определённым образом, может быть функциональной, может быть связана с пользовательским интерфейсом, или чисто технической (например-описывать как должен работать продукт изнутри). Спецификация нужна чтобы вся команда одинаково понимала какие требования к продукту существуют и каким продукт должен получится в конечном итоге. Это определение слишком размывчато, на самом деле есть большая разновидность требований и совокупность **всех** требований и есть спецификация требований.

**Рефакторинг** - изменение/редактирование кода которое не ведёт к изменению функциональности а преследует цель по оптимизации самого кода (чтобы код был более чистым)

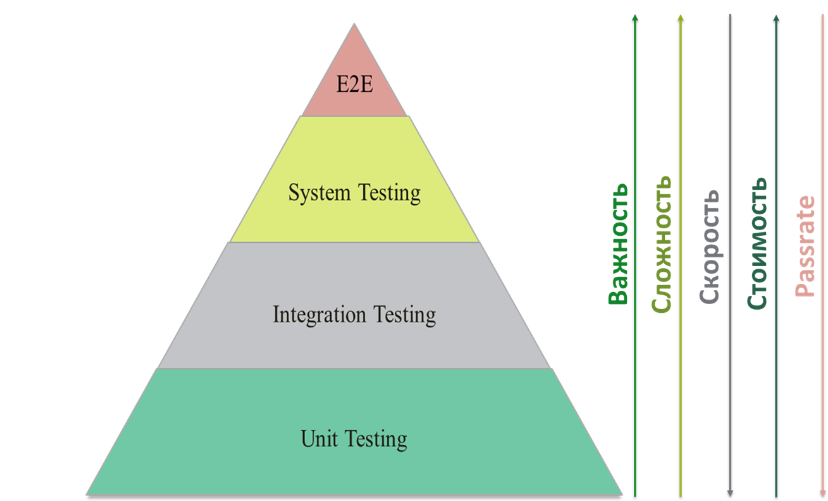
**Верификация** - доказанное объективными результатами исследования подтверждение того, что определенные требования были выполнены. [ISTQB Glossary] Верификация показывает то, что продукт разработан согласно требованиям к нему и соответствует им (нуждам заказчика). Отвечает на вопрос: правильно ли мы разрабатываем продукт? Чаще всего под верификацией подразумевают статические методы проверки (без запуска кода), которые включают в себя анализ требований, дизайна, кода.

**Валидация** - подтверждение путем экспертизы того, что рабочий продукт соответствует потребностям заинтересованной стороны. [ISTQB Glossary] показывает то, что продукт разработан согласно ожиданиям конечного пользователя (нуждам пользователя). Отвечает на вопрос: правильный ли продукт мы разработали? Валидация включает в себя динамические проверки (с запуском кода) и тестирование уже рабочего продукта.

Пример: Пользователь оставляет отзыв на приложение в маркете

**Пирамида тестирования**

«Пирамида тестов» - метафора, которая означает группировку динамических тестов программного обеспечения по разным уровням. Она также дает представление, какое количество тестов должно быть в каждой из этих групп. Основной принцип разделения уровней - тест должен быть на том же уровне, что и тестируемый объект. В тесте более высокого уровня вы не тестируете всю условную

логику и пограничные случаи, которые уже покрыты тестами более низкого уровня. 

Уровни тестирования:

* Unit/component/program/module testing - тестируется минимально-атомарный модуль программы, чаще всего это одна функция или метод. Таких тестов должно быть больше всего;
* Integration testing - несколько модулей программы тестируются вместе;
* System testing - вся программа тестируется полностью;
* Acceptance testing - программа принимается заказчиком на соответствие заявленным требованиям либо тестировщики проходят end-to-end сценарии с точки зрения пользователя;

Если копнуть глубже, то **end-to-end** тестирование, или сквозное тестирование - это вид тестирования, используемый для проверки программного обеспечения от начала до конца, а также его интеграцию с внешними интерфейсами. На примере интернет магазина, это одна объемная проверка, которая включает в себя все шаги пользователя, начиная от регистрации в системе и заканчивая покупкой товара в магазине.