

## Глоссарий курса

### “Тестировщик программного обеспечения: с нуля до первых проектов”

**Автоматизированное тестирование** — набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования.

**API (Application Programming Interface)** — описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой.

**Баг (дефект)** — недостаток компонента или системы, который может привести к отказу определенной функциональности.

**Приоритет багов** — важность той или иной ошибки в ПО:

- Trivial — косметическая малозаметная проблема.
- Minor — очевидная, незначительная проблема.
- Major — значительная проблема.
- Critical — проблема, нарушающая работу с ключевыми функциями ПО.
- Blocker — проблема, нарушающая функционирование ПО.

**Баг-репорт** — документ, описывающий ситуацию или последовательность действий приведшую к некорректной работе объекта тестирования, с указанием причин и ожидаемого результата.

**Валидация** — определение соответствия разрабатываемого ПО ожиданиям и потребностям пользователя, требованиям к системе.

**Верификация** — процесс оценки системы или её компонентов с целью определения удовлетворяют ли результаты текущего этапа разработки условиям, сформированным в начале этого этапа.

**Веб-приложение** — клиент-серверное приложение, в котором клиентом выступает браузер, а сервером — веб-сервер.

**Жизненный цикл ПО** – период времени, который начинается с момента принятия решения о необходимости создания ПО и заканчивается в момент его полного изъятия из эксплуатации.

**Клиент-серверная архитектура** - вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами.

**Модель разработки ПО** - структура, систематизирующая различные виды проектной деятельности, их взаимодействие и последовательность в процессе разработки ПО.

**Спецификация** — детальное описание того, как должно работать ПО.

**Система отслеживания ошибок** (*англ. bug tracking system*) — программа учета и/или контроля багов:

- Atlassian JIRA
- Bugzilla
- YouTrack
- Redmine
- etc.

**Тестирование** — процесс проверки соответствия заявленных к продукту требований и реально реализованной функциональности, осуществляемый путем наблюдения за его работой в искусственно созданных ситуациях и на ограниченном наборе тестов, выбранных определенным образом.

**Тестировщик ПО** — специалист, который занимается тестированием программного обеспечения, поиском и устранением ошибок и сбоев в работе программы. В нашей должностной инструкции тестировщика ПО прописаны обязанности этого специалиста, среди которых: разработка планов, графиков, методик и описаний тестирования, выполнение тестирования программных продуктов.

**Обеспечение качества (Quality Assurance, QA)** — совокупность мероприятий, охватывающих все технологические этапы разработки, выпуска и эксплуатации программного обеспечения

**Отладка** (*англ. Debugging*) — процесс, позволяющий получить программное обеспечение, функционирующее с требуемыми характеристиками в заданной области входных данных.

**Ошибка** (*англ. Error*) – действие, которое порождает неправильный результат.

**Отчёт о дефекте** - документ, описывающий и приоритезирующий обнаруженный дефект, а также содействующий его устранению.

**Отчёт о результатах тестирования** - документ, обобщающий результаты работ по тестированию и содержащий информацию, достаточную для соотнесения текущей ситуации с тест-планом и принятия необходимых управленческих решений.

**Сбой** (*англ. Failure*) – несоответствие фактического результата работы компонента или системы ожидаемому результату.

**Требование** - описание того, какие функции и с соблюдением каких условий должно выполнять приложение в процессе решения полезной для пользователя задачи.

Классификация по типу тестирования:

**Мобильное тестирование** — тестирование мобильных приложений.

**Консольное тестирование** — тестирование приложений предназначенных для консолей.

**Web-тестирование** (Браузерное тестирование) — тестирование браузерных приложений.

Классификация по запуску кода на исполнение:

**Статическое тестирование** (*англ. Static testing*) — тестирование без запуска кода на исполнение.

**Динамическое тестирование** (*англ. Dynamic testing*) — тестирование с запуском кода на исполнение.

Классификация по доступу к коду и архитектуре ПО:

**Черный ящик** (*англ. Black box*) — тестировщику не известно как устроена тестируемая система.

**Белый ящик** (*англ. White box*) — тестировщику известно все детали реализации тестируемой системы.

**Серый ящик** (*англ. Grey box*) — тестировщику известно только некоторые особенности устройства тестируемой системы.

Классификация по степени автоматизации:

**Ручное тестирование** (*англ. Manual testing*) — тестирование ПО будучи его пользователем.

**Автоматизированное тестирование** (*англ. Automated testing*) — тестирование ПО при помощи специальных программ.

Классификация по принципу работы с приложением:

**Позитивное тестирование** (*англ. Positive testing*) — тестирование ПО на то, как оно должно работать.

**Негативное тестирование** (*англ. Negative testing*) — тестирование ПО на то, как оно не должно работать.

Классификация по уровню детализации приложения:

**Интеграционное тестирование** — тестирование взаимодействия и связей нескольких компонентов приложения.

**Системное тестирование** — это тестирование всего приложения от начала и до конца.

**Модульное тестирование** — тестирование на уровне отдельного функционального компонента приложения.

Классификация по целям и задачам:

**Функциональное тестирование** — проверка корректности работы функциональности приложения.

**Нефункциональное тестирование** — проверка нефункциональных особенностей приложения (удобство использования, совместимость, производительность, безопасность).

**Инсталляционное тестирование** — проверка протекания стадии инсталляции (установки) приложения.

**Регрессионное тестирование** — проверка на наличие багов, вызванных изменениями в приложении.

**Повторное тестирование** — выполнение тест-кейсов, которые ранее обнаружили дефекты, с целью подтверждения устранения дефектов.

**Приёмочное тестирование** — тестирование, направленное на проверку приложения с точки зрения конечного пользователя/заказчика

**Тестирование удобства использования** — тестирование, направленное на исследование того, насколько конечному пользователю понятно, как работать с продуктом, а также на то, насколько ему нравится использовать продукт.

**Тестирование доступности** — тестирование, направленное на исследование пригодности продукта к использованию людьми с ограниченными возможностями

**Тестирование интерфейса** — тестирование, направленное на проверку интерфейсов приложения или его компонентов.

**Тестирование безопасности** — тестирование, направленное на проверку способности приложения противостоять злонамеренным попыткам получения доступа к данным или функциям

**Тестирование интернационализации** — тестирование, направленное на проверку готовности продукта к работе с использованием различных языков и с учётом различных национальных и культурных особенностей.

**Тестирование локализации** — тестирование, направленное на проверку корректности и качества адаптации продукта к использованию на том или ином языке с учётом национальных и культурных особенностей.

**Тестирование совместимости** — тестирование, направленное на проверку способности приложения работать в указанном окружении (браузер, мобильное ус-во и т.д.).

**Тестирование данных и баз данных** — тестирование, направленное на исследование таких характеристик данных, как полнота, непротиворечивость, целостность, структурированность и т.д.

**Тестирование использования ресурсов** — совокупность видов тестирования, проверяющих эффективность использования приложением доступных ему ресурсов и зависимость результатов работы приложения от количества доступных ему ресурсов.

**Сравнительное тестирование** — тестирование, направленное на сравнительный анализ преимуществ и недостатков разрабатываемого продукта по отношению к его основным конкурентам.

**Демонстрационное тестирование** — формальный процесс демонстрации заказчику продукта с целью подтверждения, что продукт соответствует всем заявленным требованиям.

**Избыточное тестирование** — тестирование приложения со всеми возможными комбинациями всех возможных входных данных во всех возможных условиях выполнения.

**Тестирование надёжности** — тестирование способности приложения выполнять свои функции в заданных условиях.

**Тестирование восстанавливаемости** — тестирование способности приложения восстанавливать свои функции и заданный уровень производительности, а также восстанавливать данные в случае возникновения критической ситуации.

**Тестирование отказоустойчивости** — тестирование, заключающееся в эмуляции или реальном создании критических ситуаций с целью проверки способности приложения задействовать механизмы, предотвращающие нарушение работоспособности, производительности и повреждения данных.

**Тестирование производительности** — исследование показателей скорости реакции приложения на внешние воздействия при различной по характеру и интенсивности нагрузке.

**Нагрузочное тестирование** — исследование способности приложения сохранять заданные показатели качества при нагрузке в допустимых пределах и некотором превышении этих пределов/

**Тестирование масштабируемости** — исследование способности приложения увеличивать показатели производительности в соответствии с увеличением количества доступных приложению ресурсов.

**Объёмное тестирование** — исследование производительности приложения при обработке различных (как правило, больших) объёмов данных.

**Стрессовое тестирование** — исследование поведения приложения при нештатных изменениях нагрузки, значительно превышающих расчётный уровень.

**Конкурентное тестирование** — исследование поведения приложения в ситуации, когда ему приходится обрабатывать большое количество одновременно поступающих запросов, что вызывает конкуренцию между запросами за ресурсы (базу данных, память, канал передачи данных, дисковую подсистему и т.д.)

**Фокус-тест** (*англ. Focus test*) — тестирование, проводимое с целью получения первичной реакции игроков. Необходимо для оценки удобства использования и того, как продукт принимается целевой аудиторией или сторонними людьми.

**Failure** — сбой (причём не обязательно аппаратный) в работе компонента, всей программы или системы.

**UX** (*англ. User eXperience — опыт пользователя*) — ощущение, испытываемое пользователем во время использования цифрового продукта.

**UI** (*англ. User Interface — пользовательский интерфейс*) — это инструмент, позволяющий осуществлять взаимодействие «пользователь — приложение».

**Анализ граничных значений** (*англ. Boundary Value Analysis — BVA*).

Анализ граничных значений может быть применен к полям, записям, файлам, или к любого рода сущностям имеющим ограничения.

**Дымовое тестирование** (*англ. Smoke test*) — короткий цикл тестов для подтверждения, что после сборки кода (нового или исправленного) приложение стартует и выполняет основные функции.

**Исследовательское (ad-hoc) тестирование** — это разработка и выполнения тестов в одно и то же время, что является противоположностью сценарного подхода.

**Конфигурационное тестирование** (*англ. 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.)

**Матрица соответствия требований** (*англ. Traceability matrix*) — это двумерная таблица, содержащая соответствие функциональных требований (functional requirements) продукта и подготовленных тестовых сценариев (test cases).

**Операционное тестирование** (*англ. Release Testing*). Даже если система удовлетворяет всем требованиям, важно убедиться в том, что она удовлетворяет нуждам пользователя и выполняет свою роль в среде своей эксплуатации, как это было определено в бизнес модели системы.

**Предугадывание ошибки** (*англ. Error Guessing — EG*). Это когда тест аналитик использует свои знания системы и способность к интерпретации спецификации на предмет того, чтобы «предугадать» при каких входных условиях система может выдать ошибку.

**Причина / Следствие** (*англ. Cause/Effect — CE*). Это, как правило, ввод комбинаций условий (причин), для получения ответа от системы (Следствие).

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

**Серьезность** (*англ. Severity*) — это атрибут, характеризующий влияние дефекта на работоспособность приложения.

**Стадии разработки ПО** — это этапы, которые проходят команды разработчиков ПО, прежде чем программа станет доступной для широко круга пользователей.

**Пре-альфа** (*англ. Pre-alpha*) — начальная стадия разработки. Период времени со старта разработки до выхода стадии Альфа. Также так называются программы, прошедшие стадию разработки, для первичной оценки функциональных возможностей в действии.

**Альфа-тестирование** (*англ. Alpha testing*) — имитация реальной работы с системой штатными разработчиками, либо реальная работа с системой потенциальными пользователями/заказчиком на ранней стадии разработки продукта, но в некоторых случаях может применяться для законченного продукта в качестве внутреннего приёмочного тестирования.

**Бета-тестирование** (*англ. Beta testing*) — интенсивное использование почти готовой версии продукта с целью выявления максимального числа ошибок в его работе для их последующего устранения перед окончательным выходом (релизом) продукта на рынок, к массовому потребителю.

**Релиз-кандидат или RC** (*англ. Release candidate*), Пре-релиз, иногда «гамма-версия» — стадия-кандидат на то, чтобы стать стабильной.

**Релиз или RTM** (*англ. Release to manufacturing — промышленное издание*) — издание продукта, готового к тиражированию.

**Пост-релиз или Post-RTM** (*англ. Post-release to manufacturing*) — издание продукта, у которого есть несколько отличий от RTM и помечается как самая первая стадия разработки следующего продукта.

**Таблица принятия решений** (*англ. Decision table*) — инструмент для упорядочения сложных бизнес требований, которые должны быть реализованы в продукте.

**Тест-дизайн** (*англ. Test design*) — это этап процесса тестирования ПО, на котором проектируются и создаются тестовые случаи (тест кейсы).

**Тест-план** (*англ. Test Plan*) — это документ, описывающий весь объем работ по тестированию, а также оценки рисков с вариантами их разрешения.

**Тестирование взаимодействия** (*англ. Interoperability Testing*) — это функциональное тестирование, проверяющее способность приложения взаимодействовать с одним и более компонентами или системами.

**Тестирование сборки** (*англ. Build Verification Test*) — тестирование направленное на определение соответствия, выпущенной версии, критериям качества для начала тестирования.

**Тестирование пользовательского интерфейса** (*англ. UI Testing*) — тестирование, выполняемое с целью определения, удобен ли некоторый искусственный объект (такой как веб-страница, пользовательский интерфейс или устройство) для его предполагаемого применения.



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

**Чек-лист** (*англ. Check list*) — это документ, описывающий что должно быть протестировано.

**Эквивалентное Разделение** (*англ. Equivalence Partitioning — EP*). Как пример, у вас есть диапазон допустимых значений от 1 до 10, вы должны выбрать одно верное значение внутри интервала, скажем, 5, и одно неверное значение вне интервала — 0.