Итак, какие артефакты необходимы для обеспечения процесса тестирования (имеется ввиду разрабатываемые самим тестировщиком или тестинг-менеджерами).  
  
1. Test plan - План тестирования;

2. Test Case - Тестовый сценарий;

3. Automated Test Cases;

4. Defects description;

5. Summary report. – генерируется в конце фазы тестирования и содержит в себе информацио о том:

- сколько ТК было выполнено;

- соклько их них были спашены;

- сколько ТК были сфэйлены.

**Test Plan**

Это документ, который содержит идею и описание стратегии тестирования в целом:

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.

Эти условия включают:

- тип и уровни тестирования;

- использование автоматизированного тестирования;

- любые специальные инструменты;

- настройки конфигурации среды;

- специальные требования.

Имеются шаблоны **тест планов** от [**RUP (Rational Unified Process)**](http://alexeybulat.blogspot.com/2008/01/rup-project-process.html) и **стандарт IEEE 829**:

Хороший тест план должен отвечать на такие вопросы:

1. **Что нужно тестировать?** описание объекта тестирования: системы, приложения, оборудование; список функций и описание тестируемой системы и её компонент в отдельности
2. **Как будете тестировать?** стратегия тестирования, а именно: [виды тестирования](http://www.protesting.ru/testing/testtypes.html) и их применение по отношению к тестируемому объекту
3. **Когда будете тестировать?** последовательность проведения работ: подготовка (Test Preparation), тестирование (Testing), анализ результатов (Test Result Analisys ) в разрезе запланированных фаз разработки
4. **Критерии начала тестирования:**

готовность энвармента (тестового стенда)

законченность разработки требуемого функционала

наличие всей необходимой документации

...

1. **Критерии окончания тестирования:**

результаты тестирования удовлетворяют критериям качества продукта

[требования к количеству открытых багов](http://www.protesting.ru/testing/bugpriority.html#openbugsreq) выполнены

**Test Case**

Согласно стандартному глоссарию терминов of Software Engineering (IEEE 610) test case is:

* A set of input values, execution preconditions, expected results and execution, post conditions, developed for a particular objective (цель) or test condition, such as to exercise a particular program path or to verify compliance with a specific requirement.
* Documentation specifying inputs, predicted results, and a set of execution conditions for a test item.

Т. е. это набор входных данных, начальных условий, ожидаемых результатов, конечных условий, разработанных для проверки некоторой функциональности программы на ее соответствие рекварментам. **Тестовый случай** (Test Case) - это совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.

Обычно тест кейсы объединяются в **Тест Сеты**. Если эти Тест Сеты слишком большие, то они разбиваются на более маленькие тест сеты.

**Для чего пишутся тест кейсы?**

* Early bug detection. Тестовые сценарии могут быть проревьювлены коллегами и разработчиками. Ошибки в коде и тестов могут быть предотвращены до конца стадии разработки, и могут появляться новые идеи для написания тест кейсов. Идея состоит в написании тестов во время написания кода, для того чтобы мы тестирование продукт произвести быстро, когда разработка будет завершена.
* Legal documents of testing work, in case information is needed for law suits. Тст кейсы являются необходимой документацией для предостваления отчетности.
* Ability to track history while iterations. Возможность просматривать развитие каждой версии.
* Ability to measure the percentage of testing done. Возможность оценивать процент выполненой работы.
* Traceability to requirements. Просмотр требований
* Usefulness while bringing in new testers. Это удобно для ньюкамеров.

**Attributes**

**Type**: Reflects the type of test case depending on what kind of testing is covered by particular test case: Manual, Automated etc.

**Criticality**: критичность с точки зрения безопасности работы программы: Critical, Non-Critical

# Best Practices

## Details required before writing test cases

* Read Requirements and Specifications carefully
* Be clear with design and implementation details
* Analyze all possible scenarios in Test Design Specification
* Be clear of behavior under failure condition (invalid inputs, boundary condition)
* Define standards for writing test cases

## Qualities of good test case

* Easy and quick to determine the result
* The test should need no more than a few seconds to convey their results to tester
* The test should not need an understanding of the requirements to be used
* Short (up to 10 steps)

Note:

Тест Кейсы могут быть более или менее сложными, более или менее подробными. В реальных условиях контент ТК , скорее всего, будет разным, в зависимости от вашего проекта. Подходы для написния ТК варьируются от самых подробных описаний шагов до написания каких-либо общих инструкций.

Большинство организаций предпочитают подробные тесты. Кроме того, подробные тесты являются легче воспроизводимыми, и их легче автоматизировать, чем глобальные тест-кейсы. Однако детальные тесты требуют больших временных затрат для их разработки и поддержки.

## The seven most common mistakes

* Making cases too long
* Incomplete, incorrect, or incoherent (несвяханные) test case steps
* Leaving out a step
* Naming fields that changed or no longer exist
* Unclear what the next action is done by user / system
* Unclear expected result (what is a pass or fail criteria)

**Test Case Example**

**Требования:** пользователь запускает приложение, выбирает опцию «Добавить нового пользователя». Открывается форма, он заполняет поля, нажимает Save. Появляется сообщение «Поздравляем! Пользователь был добавлен». Сообщение закрывается.

**Test Case Practice**

Подумайте, какие условия в рекварменты вы бы добавили.

**Summary Report**

В саммари репорте отражаются статусы всех пройденных или сфэйленных ТК и описываются причины, если это фэйл:

- pass: для каждого ТК все степы должны быть успешно пройдены. Степ любого ТК со статусом «no run» должен иметь объяснение причины, почему шаг был не пройден.

- fail: каждый степ должен быть выполнен. Любому результату, который отличается от ожидаемого, должен быть описан и в дальнейшем оценен. Любой статус «no run» должен иемть объяснение.

В ISD Summary Report генерируется тестинг менеджерами.

**Test Case Management Tools**

* Management of tests (например, отслеживание соответствующих данных для данного набора тестов, зная, **какие** тесты нужно запускать, **количество** тестов, которые **planned, written, run, passed or failed**)
* Scheduling of tests to be executed (вручную или с помощью инструмента выполнениятеста);
* Management of testing activities (время, затраченное на проектирование тестов,выполнение теста, укладываемся ли мы в график бюджета)
* Interfaces to other tools.

**Features:**

* + Ability to create new and effectively manage existing Test Cases
  + Easy to maintain Test Cases
  + Ability to track everything about a test (test case details, test steps, variants, file attachments, test instructions, test runs, workflow, email, links to other tests, folders, and its history)
  + Ability to organize and categorize your Test Cases by Product, Component, Test Type, Test Component and Test Subcomponent
  + Versioning of Test Cases
  + Group Test Cases into Test Sets
  + Presence of search capabilities
  + Track Test Case execution times, total run time, and estimate workload
  + Ability to link Test Cases with requirements and vice versa
  + Traceability Matrix generation
  + Ability to create reports, etc

**Requirements** A condition or capability needed by a user to solve a problem or achieve an

objective that must be met or possessed by a system or system component to satisfy a

contract, standard, specification, or other formally imposed document. [After IEEE 610]

Условия или возможности, необходимые пользователю для решения каких-либо задач или достижения цели, которые должны быть выполнены или достигнуты системой или её компонентом для удовлетворения контракт, стандарт, спецификации, или других официальных документов.

**Требования к программному обеспечению** — совокупность утверждений относительно атрибутов, свойств или качеств программной системы, подлежащей реализации.

**specification:** A document that specifies, ideally in a complete, precise and verifiable manner,

the requirements, design, behavior, or other characteristics of a component or system, and,

often, the procedures for determining whether these provisions have been satisfied. [After

IEEE 610]

документ, который наиболее полно и точно определяет точной и поддающимся проверке образом, требований, проектирования, поведение или другие характеристики элемента или системы, и, часто, процедуры для определения, являются ли эти положения были удовлетворены.

Спецификации (часто сокращенно спецификации) является явной набор требований и параметров, которым должны удовлетворять материал, продукт или услугу.

User Story - пособ описания требований к разрабатываемой системе, сформулированных как одно или более предложений на ежедневном или деловом языке пользователя.

**Прецеде́нт** ([англ.](http://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Use Case*), также: **вариант использования**, [**сценарий использования**](http://ru.wikipedia.org/wiki/%D0%A1%D1%86%D0%B5%D0%BD%D0%B0%D1%80%D0%B8%D0%B9_%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) — [спецификация](http://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F) последовательностей действий (варианты последовательностей и ошибочные последовательности) в Унифицированном языке моделирования ([UML](http://ru.wikipedia.org/wiki/UML)), которые может осуществлять система, подсистема или класс, взаимодействуя с внешними [акторами](http://ru.wikipedia.org/wiki/%D0%AD%D0%BA%D1%82%D0%BE%D1%80_(UML)) ([англ.](http://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Actors*).

\\isd103\Documentation\SDLC\_course