

SATEL

SATEL

Тестовая документация



# Что такое дефект (баг)?

Предельно просто

**Дефект (defect, bug, problem, fault)** — несоответствие фактического и ожидаемого результата.

**Ожидаемый результат** — поведение системы, описанное в требованиях.

**Фактический результат** — поведение системы, наблюдаемое в процессе тестирования.

Более подробно

**Дефект** — недостаток в компоненте или системе, способный привести к ситуации сбоя или отказа.

**Сбой** — самоустраняющийся отказ или однократный отказ, устраняемый незначительным вмешательством оператора.



**Отказ** — событие, заключающееся в нарушении работоспособного состояния объекта.

## Атрибуты баг репортов

- **Идентификатор** представляет собой уникальное значение, позволяющее однозначно **отличить один отчёт о дефекте от другого и используемое во всевозможных ссылках.**
- **Краткое описание (Summary)** — должно в предельно лаконичной форме давать исчерпывающий ответ на вопросы: «Что произошло?» «Где это произошло»? «При каких условиях это произошло?»
- **Подробное описание** представляет в развёрнутом виде необходимую информацию о дефекте, а также описание фактического результата, ожидаемого результата и ссылку на него.
- **Шаги воспроизведения максимально подробно.**
- **Воспроизводимость** показывает, при каждом ли прохождении по шагам воспроизведения дефекта удаётся вызвать его проявление. Имеет всего два значения:

**всегда (always) или иногда (sometimes).**

## Атрибуты баг репортов (Продолжение)

- **Серьёзность (severity)** — это то, как баг влияет на систему.
- **Приоритет (priority)** — это то, как быстро баг починится.
- **Автор**
- **Кто будет чинить этот дефект?** На кого назначен?
- **Окружение** -**DEV окружение** – разработчики; -**STAGE окружение** – тестировщики; -**PROD окружение** - конечные пользователи.
- **Комментарий** может содержать любые полезные для понимания и исправления дефекта данные.
- **Возможность обойти** — показывает, существует ли альтернативная последовательность действий, выполнение которой позволило бы пользователю достичь поставленной цели.
  - **Прикрепления** (скриншоты, записи видео и пр.).
  - **Симптом** — позволяет классифицировать дефекты по их типичному проявлению.



# Приоритет баг репортов

Показывает, как быстро дефект должен быть устранён.

**Наивысшая (ASAP - as soon as possible)** срочность указывает на необходимость устранить дефект настолько быстро, насколько это возможно.

**Высокая (high)** срочность означает, что дефект следует исправить вне очереди, т.к. его существование или уже объективно мешает работе, или начнёт создавать такие помехи в самом ближайшем будущем.

**Обычная (normal)** срочность означает, что дефект следует исправить в порядке общей очерёдности. Такое значение срочности получает большинство дефектов.

**Низкая (low)** срочность означает, что в обозримом будущем исправление данного дефекта не окажет существенного влияния на повышение качества продукта.

## Серьёзность (severity)

Показывает степень ущерба, который наносится проекту существованием дефекта.

- A. **Blocker** - такой тип ошибки, который приводит нашу программу в **нерабочее состояние**.
- B. **Критическая (critical)** — существование дефекта приводит к **масштабным последствиям катастрофического характера**.
- C. **Высокая (major)** — существование дефекта приносит **ощутимые неудобства многим пользователям** в рамках их типичной деятельности.
- D. **Средняя (medium)** — существование дефекта **слабо влияет на типичные сценарии работы пользователей**, и/или существует обходной путь достижения цели.
- E. **Низкая (minor)** — существование **дефекта редко обнаруживается незначительным процентом пользователей и (почти) не влияет на их работу**.



F. **Trivial** - какие-то тривиальные баги.

## СИМПТОМ

Позволяет классифицировать дефекты по их типичному проявлению.

- A. Косметический дефект**
- B. Повреждение/потеря данных**
- C. Проблема в документации**
- D. Некорректная операция**
- E. Проблема инсталляции**
- F. Ошибка локализации**
- G. Нереализованная функциональность**
- H. Проблема масштабируемости**
- I. Низкая производительность**
- J. Крах системы**
- K. Неожиданное поведение**
- L. Недружественное поведение**
- M. Расхождение с требованиями**
- N. Предложение по улучшению**

# Жизненный цикл багов и баг репортов

- **Обнаружен (submitted)** — начальное состояние отчёта, в котором он находится сразу после создания.
- **Назначен (assigned)** — в это состояние отчёт переходит с момента, когда кто-то из проектной команды назначается ответственным за исправление дефекта.
- **Исправлен (fixed)** — в это состояние отчёт переводит ответственный за исправление дефекта член команды после выполнения соответствующих действий по исправлению.
- **Проверен (verified)** — в это состояние отчёт переводит тестировщик, удостоверившись, что дефект на самом деле был устранён.
- **Закрыт (closed)** — состояние отчёта, означающее, что по данному дефекту не планируется никаких дальнейших действий.





## Возможные состояния баг репортов

- **Открыт заново (reopened)** — в это состояние (как правило, из состояния «Исправлен»)

отчёт переводит тестировщик, удостоверившись, что дефект по прежнему воспроизводится на билде, в котором он уже должен быть исправлен.

- **Рекомендован к отклонению (to be declined)** — в это состояние отчёт о дефекте может

быть переведён из множества других состояний с целью вынести на рассмотрение вопрос об отклонении отчёта по той или иной причине.

- **Отклонён (declined)** — в это состояние отчёт переводится в случаях, подробно описанных в

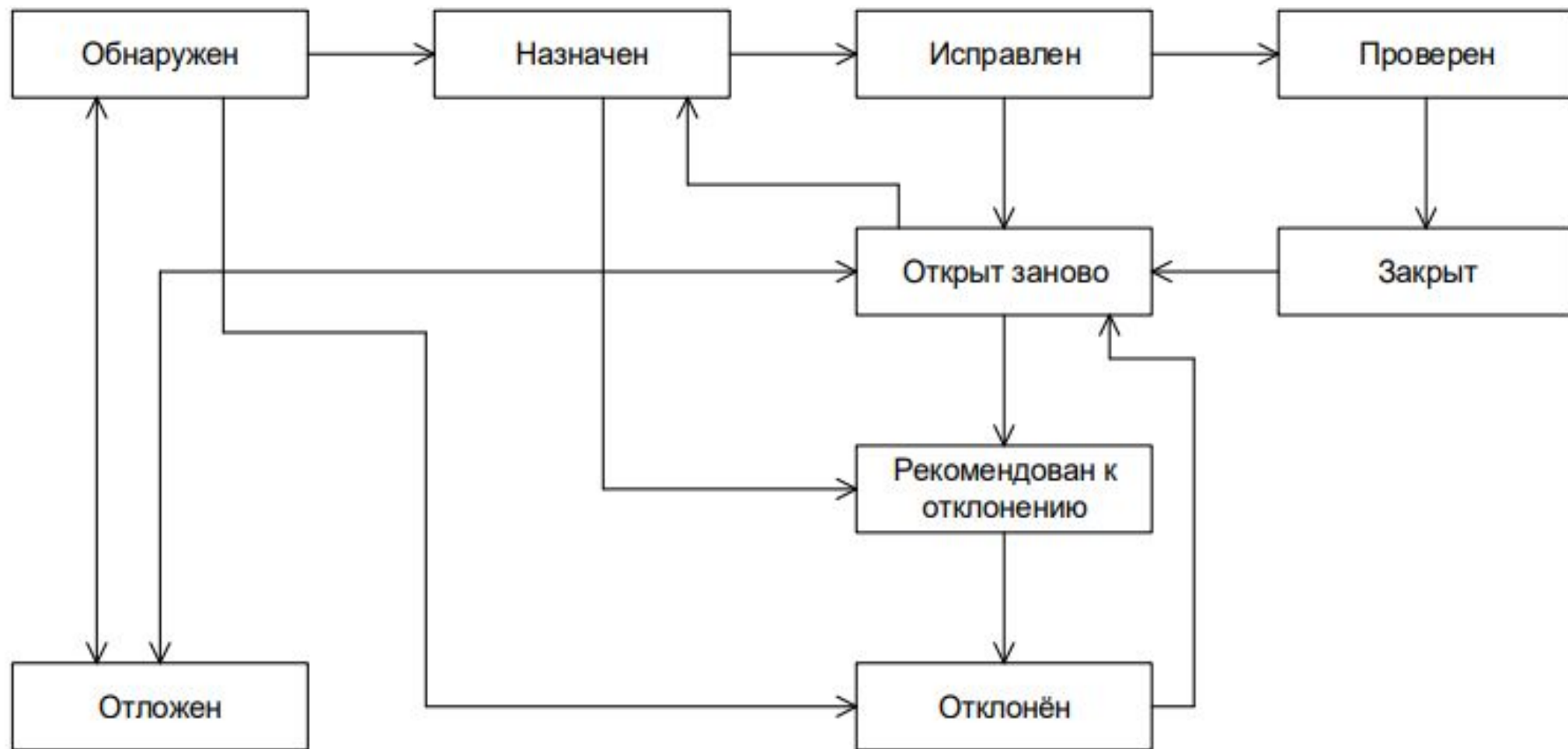
пункте «Закрыт», если средство управления отчётами о дефектах предполагает использование этого состояния вместо состояния «Закрыт» для тех или иных резолюций по отчёту.

- **Отложен (deferred)** — в это состояние отчёт переводится в случае, если

исправление дефекта в ближайшее время является нерациональным или не представляется возможным, однако есть основания полагать, что в будущем ситуация исправится.



# Жизненный цикл бага и баг репорта наглядно



# Чек-листы

В общем случае **чек-лист** — это просто **набор идей**: идей по тестированию, идей по разработке, идей по планированию и управлению — любых идей.

Чек-лист чаще всего **представляет собой** обычный и привычный нам **список**:

- в котором **последовательность пунктов не имеет значения**;
- в котором **последовательность пунктов важна**;
- **структурированный (многоуровневый)**.

Для того чтобы чек-лист был действительно полезным инструментом, он должен обладать рядом

**важных свойств**:

- Логичность**;
- Последовательность и структурированность**;
- Полнота и избыточность**.



## Атрибуты чек-листа

- **Версия** нашей сборки (**билд**).
- **Окружение (environment)**, на котором проводилось тестирование.
- **Дата** проведения нашего теста.
- **Тестировщик**, который проводил данное тестирование.
- **Тип тестов**, который был использован для тех или иных проверок.
- **Название** самих проверок.
- **Результат** тестирования, т.е. прошла эта проверка или нет.



# Тест-кейсы

**Тест-кейс** — набор входных данных, условий выполнения и ожидаемых результатов, разработанный с целью проверки того или иного свойства или поведения программного средства.

В отличие от чек-листа, в котором мы говорили, **ЧТО** мы будем тестировать, здесь мы уже расписываем **КАК** мы будем тестировать.

**Высокоуровневый тест-кейс** — тест-кейс **без конкретных входных данных и ожидаемых результатов.**

**Низкоуровневый тест-кейс** — тест-кейс **с конкретными входными данными и ожидаемыми результатами.**

**Спецификация тест-кейса** — документ, описывающий набор тест-кейсов для тестируемого элемента.



**Спецификация теста** — документ, состоящий из спецификации тест-дизайна, спецификации тест-кейса и/или спецификации тест-процедуры.

## Цели написания тест-кейсов:

- A. Структурировать** и систематизировать подход к тестированию.
- B.** Вычислять метрики **тестового покрытия** и принимать меры по его увеличению.
- C. Отслеживать соответствие текущей ситуации плану.**
- D. Уточнить взаимопонимание** между заказчиком, разработчиками и тестировщиками.
- E. Хранить информацию** для длительного использования и обмена опытом между сотрудниками и командами.
- F. Проводить регрессионное тестирование** и повторное тестирование.
- G. Повышать качество требований.**
- H. Быстро вводить в курс дела** нового сотрудника, недавно подключившегося к проекту.

## Жизненный цикл тест-кейса:

- **Создан**

- **Запланирован** — тест-кейс **или явно включён в план ближайшей итерации** тестирования, или готов для выполнения.

- **Не выполнен** — нахождение тест-кейса в данном состоянии означает, что он **готов к выполнению, но ещё не был выполнен.**

- **Пропущен** — бывают ситуации, когда выполнение тест-кейса отменяется.

- **Выполняется**



- **Провален** — данное состояние означает, что в процессе выполнения тест-кейса был обнаружен дефект

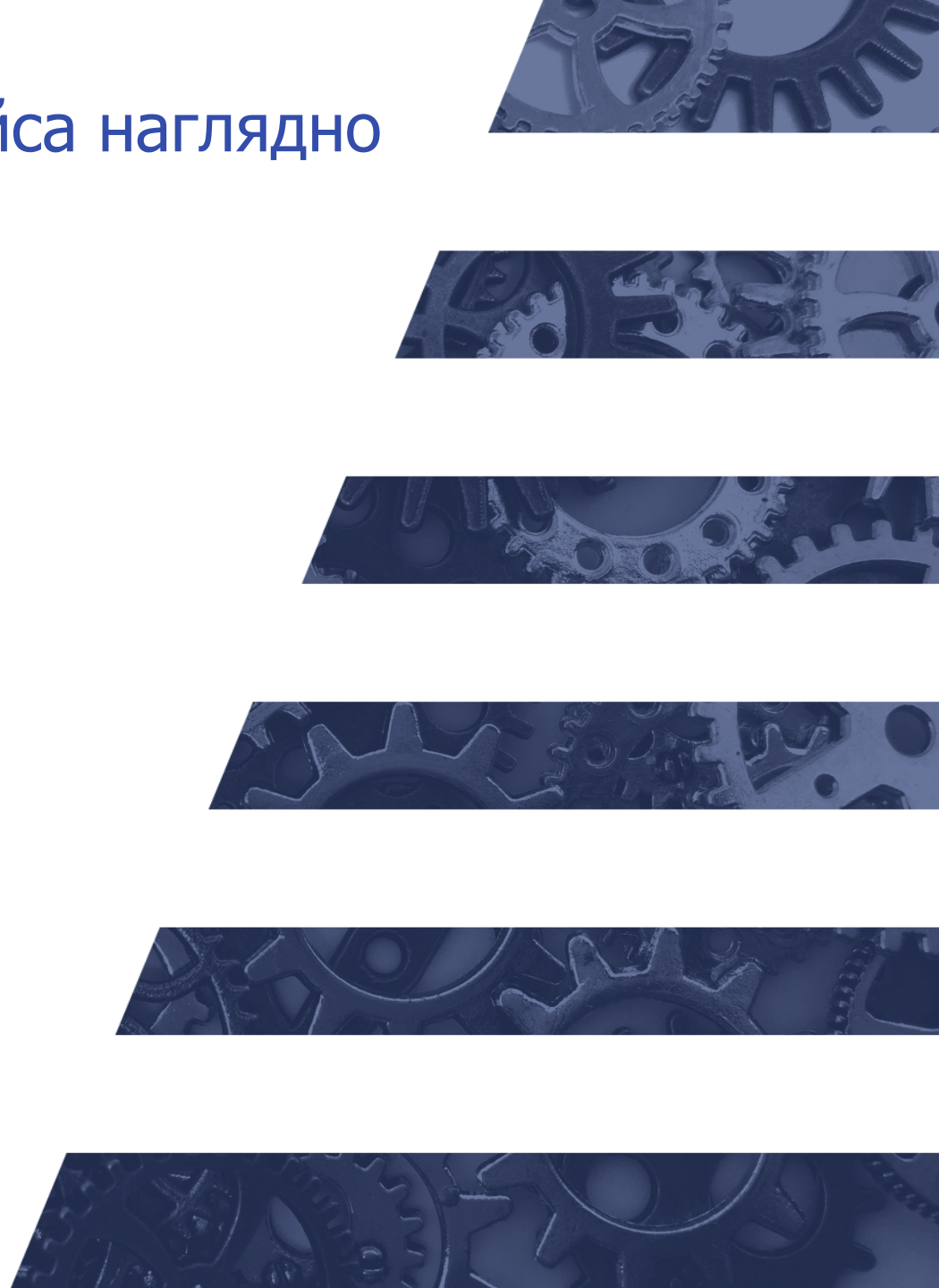
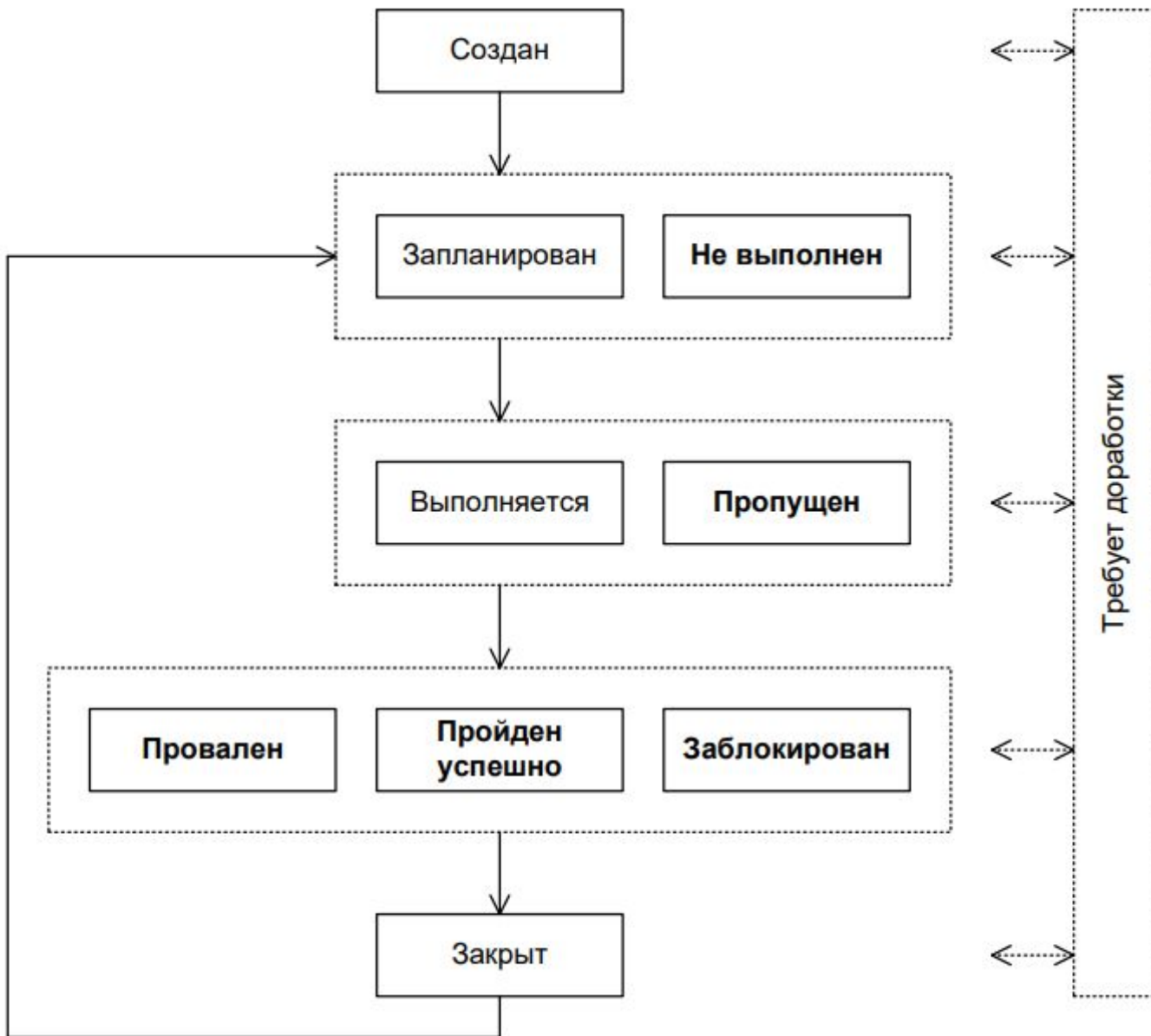
- **Пройден успешно** — данное состояние означает, что в процессе выполнения тест-кейса не было обнаружено дефектов

- **Заблокирован** — данное состояние означает, что **по какой-то причине выполнение тест-кейса невозможно**

- **Закрыт**

- **Требует доработки**

# Жизненный цикл тест-кейса наглядно





# Атрибуты тест-кейса

- 1. Идентификатор** представляет собой **уникальное значение**, используемое во всевозможных ссылках.
- 2. Приоритет** показывает **важность тест-кейса**. Выражается буквами, цифрами, словами. Количество градаций не фиксировано.
- 3.** Связанное с тест-кейсом **требование** показывает то **основное требование**, проверке выполнения которого посвящён тест-кейс.
- 4. Модуль и подмодуль приложения** указывают на части приложения, **к которым относится тест-кейс**.
- 5. Заглавие тест-кейса** призвано упростить и ускорить понимание основной идеи.
- 6. Исходные данные**, необходимые для выполнения тест-кейса позволяют описать всё то, что должно быть подготовлено до начала выполнения тест-кейса.
- 7. Шаги тест-кейса** последовательность действий при выполнении.
- 8. Ожидаемые результаты.**

# Тест-план

**Тест-план (test plan)** — документ, **описывающий** и **регламентирующий перечень работ** по тестированию, а также соответствующие техники и подходы, стратегию, области ответственности, ресурсы, расписание и ключевые даты.

Качественный тест-план обладает большинством свойств качественных требований , а также расширяет их набор следующими пунктами:

- **Реалистичность**
- **Гибкость**
- **Согласованность** с общим проектным планом

## Разделы тест плана

- **Цель (purpose)**. Предельно краткое описание цели разработки приложения
- **Области, подвергаемые тестированию (features to be tested)**. Перечень функций и/или нефункциональных особенностей приложения, которые будут подвергнуты тестированию.
- **Области, не подвергаемые тестированию (features not to be tested)**. Перечень функций и/или нефункциональных особенностей приложения, которые не будут подвергнуты тестированию.
- **Тестовая стратегия (test strategy) и подходы (test approach)**. Описание процесса тестирования с точки зрения применяемых методов, подходов, видов тестирования, технологий, инструментальных средств и т.д.
- **Ресурсы (resources)**. В данном разделе тест-плана перечисляются все необходимые для успешной реализации стратегии тестирования ресурсы.
- **Расписание (test schedule)**. Фактически это календарь, в котором указано, что и к какому моменту должно быть сделано.



## Разделы тест плана (Продолжение)

- **Роли и ответственность (roles and responsibility).** Перечень необходимых ролей и область ответственности специалистов, выполняющих эти роли.
- **Оценка рисков (risk evaluation).** Перечень рисков, которые с высокой вероятностью могут возникнуть в процессе работы над проектом. По каждому риску даётся оценка представляемой им угрозы и приводятся варианты выхода из ситуации.
- **Документация (documentation).** Перечень используемой тестовой документации с указанием, кто и когда должен её готовить и кому передавать.
- **Метрики (metrics).** Числовые характеристики показателей качества, способы их оценки, формулы и т.д. На этот раздел, как правило, формируется множество ссылок из других разделов тест-плана.

# Критерии (criteria) тест-плана

Этот раздел включает следующие подразделы:

- **Приёмочные** критерии, критерии **качества (acceptance criteria)** — любые объективные показатели качества, которым разрабатываемый продукт должен соответствовать с точки зрения заказчика или пользователя, чтобы считаться готовым к эксплуатации.

- Критерии **начала** тестирования (**entry criteria**) — перечень условий, при выполнении которых команда приступает к тестированию.

- Критерии **приостановки** тестирования (**suspension criteria**) — перечень условий, при выполнении которых тестирование приостанавливается.

- Критерии **возобновления** тестирования (**resumption criteria**) — перечень условий, при выполнении которых тестирование возобновляется.

- Критерии **завершения** тестирования (**exit criteria**) — перечень условий, при выполнении которых тестирование завершается.



# Отчёт о результатах тестирования

**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.

Качественный отчёт о результатах тестирования обладает многими свойствами качественных требований, а также расширяет их набор следующими пунктами:

- **Информативность** (в идеале после прочтения отчёта не должно оставаться никаких открытых вопросов о том, что происходит с проектом в контексте качества).
- **Точность и объективность** (ни при каких условиях в отчёте не допускается искажение фактов, а личные мнения должны быть подкреплены твёрдыми обоснованиями).

# Атрибуты отчёта о результатах тестирования

- **Краткое описание (summary).** В предельно краткой форме отражает основные достижения, проблемы, выводы и рекомендации.
- **Команда тестировщиков.** Список участников проектной команды, задействованных в обеспечении качества, с указанием их должностей и ролей в подотчётный период.
- **Описание процесса тестирования.** Последовательное описание того, какие работы были выполнены за подотчётный период.
- **Расписание.** Детализированное расписание работы команды тестировщиков.
- **Список новых дефектов.** Список обнаруженных за подотчётный период дефектов с их краткими описаниями и важностью.
- **Статистика по всем дефектам.** Таблица, в которой представлены данные по обнаруженным за всё время существования проекта дефектам (с классификацией по стадии жизненного цикла и важности).
- **Рекомендации.** Обоснованные выводы и рекомендации по принятию тех или иных управленческих решений.
- **Приложения.** Фактические данные (как правило, значения метрик и графическое представление их изменения во времени)





**СПАСИБО!**

