

Тест-дизайн и тест-аналитика
Урок 8

Составление отчёта о тестировании





Оглавление

На этом уроке	3
Отчёт о результатах тестирования	3
Содержание отчёта о тестировании фичи	4
Содержание отчёта о тестировании задачи	6
Содержание отчёта об исследовательском тестировании	7
Частота формирования отчёта	7
Метрики и виды метрик	8
Прямые метрики	9
Расчётные метрики	9
Критерии тестирования	11
Контрольные вопросы	13



На этом уроке

1. Научимся составлять отчет о результатах тестирования.
2. Познакомимся с метриками и их видами.

Отчёт о результатах тестирования

Человек по своей натуре довольно ленив. Например, на вопрос «какой прогресс выполнения задачи на текущий момент?» многие ответят «всё хорошо», «в процессе», «выполнил 30 тест-кейсов, осталось 20». Но эти ответы не очень информативны. Что значит «всё хорошо»? Всё протестировано, и функция работает в соответствии с ТЗ? Как много сделано? Когда планируется закончить? Есть ли что-то, что мешает? Какие именно тесты были пройдены? Эти вопросы будут задавать все менеджеры и руководители.

Если недостаточно сказать «всё хорошо», то что нужно добавить в отчёт о тестировании? Давайте разбираться.

Отчёт о тестировании показывает, в какой стадии тестирования находится функционал. Что именно добавить в отчёт, зависит от заказчика.

Отчётность — это сбор и распространение информации о результатах работы, включая текущий статус, оценку прогресса и прогноз развития ситуации.

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

А теперь представьте менеджера по продукту, который хочет скорее увидеть новый функционал в продакшене. Фича была спроектирована, разработка завершила свой этап и передала в тестирование. Продакт-менеджера интересует отчёт о тестировании фичи. QA готовит следующий:

Успешно пройдено 15 тест-кейсов из 20 Не прошли проверку: 3 тест-кейса Осталось пройти: 2 тест-кейса Заведено 2 баг-репорта

Отчёт может показаться вполне профессиональным: видно, сколько пройдено тестов, что осталось, сколько заведено дефектов. Но на его основании сложно



сделать вывод, можно ли релизить функционал. А почему? В качестве примера возьмём строку «Заведено 2 баг-репорта». Из неё не ясна критичность этих баг-репортов — вдруг среди них есть дефект, с которым релиз не сможет состояться. Также непонятно, куда делся ещё один непройденный тест-кейс, был ли на него заведен дефект или прямо сейчас с ним разбираются тестировщик и разработчик. Вероятно, все эти вопросы приведут к личной встрече с QA, чтобы всё прояснить.

Как лучше было бы составить отчёт? Например, так: «Тестирование функциональности на финальной стадии. Пройдено 15 тест-кейсов из 20, 3 тест-кейса не прошли проверку. Заведено 2 дефекта, один из них блокирующий, а второй можно поправить в следующем релизе. С третьим дефектом разбираюсь прямо сейчас. Осталось пройти ещё два тест-кейса». Теперь продакт-менеджер знает, что релизить нельзя, стоит подождать исправлений.

Выше приведён пример отчёта для конкретного специалиста. А что, если отчёт нужен для нескольких одновременно? Тогда можно задать себе три вопроса:

- **Кто заказчик отчёта?** Кроме менеджера по продукту, у QA Report могут быть и другие заказчики: разработчики, руководители тестирования, руководители разработки, менеджер проекта и многие другие.
- **Какую информацию нужно предоставить?** Разработчик, вероятно, заинтересован в найденных дефектах, так что корректно добавить ссылку на них в отчёт. Руководителю тестирования нужна вся информация по проведенному вами процессу. Менеджеру проекта стоит знать, в каком состоянии мы находимся (продукт, разработка, контроль качества и так далее), поэтому уместно предоставить общий отчёт о состоянии с небольшими пояснениями.
- **Как заказчики используют отчёт?** Разные специалисты делают это по-разному:
 - менеджер по продукту — для принятия решения о выпуске фичи;
 - QA Lead — для планирования возможностей специалиста по тестированию;
 - разработчик — для планирования своих задач с учётом времени на исправление ошибок.



Содержание отчёта о тестировании фичи

Практически любой отчёт о тестировании фичи, функциональности или продукта будет содержать следующие разделы:

Краткое описание — вводный раздел, в котором кратко описывается содержание отчёта, перечисляются основные числовые показатели хода тестирования, обозначаются главные выводы и даются рекомендации для дальнейшей работы. Этой части отчёта достаточно для общей оценки ситуации заинтересованными лицами (менеджерами, руководителями команд).

Команда тестировщиков — список участников проектной команды, задействованных в тестировании, с указанием их должностей и ролей в отчётный период.

Описание процесса тестирования — вся выполненная тестировщиками работа: количество написанных и пройденных тест-кейсов, число найденных дефектов, добавленные и протестированные функции.

Расписание — календарные сроки, в которые выполнены работы.

Статус активностей тестирования и прогресс по сравнению с планом тестирования — график burndown, на котором видно соотношение запланированных и проведённых работ.

Факторы, препятствующие прогрессу — критичные дефекты, недоработанная функциональность, пробелы в требованиях.

Статистика по новым дефектам — количество дефектов, обнаруженных за отчётный период, с указанием степени важности, срочности, места обнаружения и прочего.

Список новых дефектов — список дефектов, обнаруженных в отчётном периоде, с указанием их основных атрибутов.

Статистика по всем дефектам — график, который отражает динамику обнаружения дефектов на протяжении всего процесса тестирования. В итоговом отчёте по результатам тестирования указываются все обнаруженные дефекты за весь период тестирования, распределённые по выбранной классификации, например, по степени важности, времени обнаружения.



Качество объекта тестирования — экспертное заключение об уровне качества продукта.

Рекомендации — выводы по результатам тестирования и рекомендации по повышению его эффективности в будущем.

Приложения — графики, таблицы, диаграммы, демонстрирующие числовые характеристики процесса тестирования.

Отчёт о тестировании представляется в разных видах: в текстовом, табличном и графическом. Пример отчета можно посмотреть в [дополнительных материалах](#) к конспекту.

Содержание отчёта о тестировании задачи

Отчёт о тестировании конкретной задачи составить довольно просто. Представим, что перед вами задача:

The screenshot shows a Jira task page for the project 'TASK-9192'. The task title is '[iOS] просмотренные товары'. The task is in the 'DONE' status. The description details the requirement to highlight viewed items and includes a Figma link. The right sidebar shows the 'People' section with an assignee and a 'Say thank you for' button.

Проект / TASK-9192

[iOS] просмотренные товары

Edit Add comment Assign More Problems Canceled

Details

Type: Task Status: DONE (View Workflow)

Priority: Medium Resolution: Done

Affects Version/s: None Fix Version/s: None

Component/s: None

Labels: None

Платформа: iOS

Sprint:

Story Points:

Description

Выделять просмотренные товары после просмотра. Сейчас пользователю при изменении условий поиска приходится просматривать по несколько раз один и тот же товар, это неудобно

фича включается по АБ

макеты - берем в работу вариант В :

iOS <https://www.figma.com/file/gKPc25pjKEE>

Как сделать:

если карточка товара была открыта и в ней хоть что-то показали пользователю (не было ошибок), товар считается просмотренным

просмотренный товар выделяется в ленте поиска сохраненных поисках (в избранном не надо)

инфа о просмотренных хранится на клиентах

инфа о просмотренных хранится в течение 1 недели, затем товар считается непросмотренным снова. После каждого повторного просмотра факт "просмотренности" обновляется на следующую неделю

People

Assignee:

Reporter:

Tester:

Developer:

Votes:

Watchers:

Intranet

Say thank you for

Dates

Created:

Updated:

Resolved:

Collaborators

Development

И поле для комментария:



Comment

Style ▾ B I U A ▾ ▾ ▾ + ▾

Viewable by All Users

Add Cancel

Комментарий в задаче должен содержать:

- **кейсы**, которые проверили в рамках задачи (можно перечислить текстом или прикрепить ссылку на тест-ран из TMS),
- **окружение**, на котором проверили (версия операционной системы, версия приложения; если тестируем мобильное приложение — модель устройства, если веб-приложение — браузер и его версия),
- **что не смогли проверить** — не работало окружение, какие-то кейсы нельзя проверить по какой-либо причине.
- **моменты, которые стоит зафиксировать** — риски, некритичные баги, которые будут правиться после релиза и подобное.

Содержание отчёта об исследовательском тестировании

В отчёт об исследовательском тестировании входит:

- **Список протестированных разделов или модулей продукта.** В дальнейшем с его помощью можно оценить тестовое покрытие и необходимость дополнительных исследований.
- **Список дефектов:** найденных вообще или самых критических (в зависимости от того, для кого и на какой стадии тестирования делается отчёт).
- **Найденные проблемы**, вопросы, наблюдения.
- **Риски.** Важно рассказать о том, что не было протестировано и почему так произошло — функциональность не входила в объём работ, не работал сервер, не было подходящих тестовых данных и так далее.
- **Краткий вывод** по результатам тестирования в зависимости от его цели. Например, можно ли передавать продукт заказчику для ознакомления.



Частота формирования отчёта

Отчёт о результатах тестирования может формироваться каждый день, в конце каждой итерации, каждую неделю и так далее — зависит от потребностей команды проекта. При полном завершении тестирования формируется отчёт о его результатах.

Метрики и виды метрик

Процесс тестирования опирается на **расчётные показатели**, которые позволяют контролировать и оценивать ситуацию. Их можно использовать в отчётах для оценки хода тестирования и его результатов.

Показатели устанавливаются перед началом тестирования. Они описываются в тест-плане или в другой тестовой документации. В тестировании для определения таких показателей используется термин «**метрика**».

Метрика — это:

- числовая характеристика показателя качества;
- мера, позволяющая получить численное значение некоторого свойства ПО или его спецификаций;
- показатель текущего достижения поставленных целей;

Метрики собираются во время и по завершении тестирования, чтобы оценить:

- прогресс относительно запланированного графика и бюджета;
- текущее качество объекта тестирования;
- адекватность подхода к тестированию;
- эффективность активностей тестирования по достижению целей тестирования.

Типы метрик



1. **Прямые** — для определения не производятся вычисления. Используются для расчётных метрик. Примеры:

- количество разработанных тест-кейсов;
- количество найденных дефектов;
- время прохождения тест-кейсов.

2. **Расчётные** — вычисляются по формулам. Например:

- процентное соотношение выполненных/невыполненных тест-кейсов ко всем имеющимся;
- процент успешного прохождения тест-кейсов ко всему их объёму;
- процент заблокированных тест-кейсов ко всему их объёму;
- плотность распределения дефектов;
- эффективность устранения дефектов;
- распределение дефектов по важности и срочности.

Прямые метрики

Общее количество найденных дефектов — количество дефектов, найденных за весь период тестирования. Представляется с разбивкой по степени важности / срочности или по модулям, где обнаружили дефекты.

Текущее количество дефектов — количество дефектов, найденных в текущей сборке, итерации, спринте. Показывает актуальную тестовую ситуацию и динамику изменений: улучшение или ухудшение кода, процесса разработки, качества тест-кейсов.

Общее устранение дефектов — процент устранения дефектов определённого уровня важности за время существования проекта. Сколько всего дефектов исправлено за время разработки и тестирования.

Текущее устранение дефектов — процент устранения в текущем билде или итерации дефектов, обнаруженных в предыдущей сборке или итерации. Когда выходит новая версия продукта или стартует новая итерация, нужно различать дефекты, обнаруженные в предыдущей версии, и изъяны, появившиеся в актуальной.

Плотность дефектов — количество дефектов на единицу размера: на весь продукт, спринт, функцию, объём исходного кода. Метрика сигнализирует о наиболее уязвимых модулях и участках кода.



Расчётные метрики

Выполнение тест-кейсов — $\frac{\text{количество выполненных тест-кейсов}}{\text{общее количество тест-кейсов}} * 100\%$

Ключевая метрика, которая показывает прогресс тестирования. Служит для сравнения фактического прогресса с планом и для внесения корректировки в работу тестировщиков.

Успешное прохождение тест-кейсов — $\frac{\text{количество успешно выполненных тест-кейсов}}{\text{общее количество выполненных тест-кейсов}} * 100\%$

Метрика показывает качество приложения: если доля неуспешных или заблокированных тест-кейсов высокая, это свидетельствует о некачественном коде. Нужно внести изменения в процесс разработки, внедрить новые инструменты и практики взаимных проверок.

Коэффициент ретеста дефектов — показывает, какую часть времени в тестировании занимает перепроверка исправленных дефектов. Важно, чтобы большая часть рабочего времени тестировщиков не уходила на ретесты вместо проверки новой функциональности.

1 вариант — $\frac{\text{количество закрытых дефектов}}{\text{количество новых дефектов}}$

2 вариант — $\frac{\text{затраченное время на ретест}}{\text{затраченное время на тестирование новой функциональности}}$

Если времени на перепроверку исправленных дефектов уходит больше, чем на тестирование новой функциональности, должны быть приняты меры, так как это может свидетельствовать о большом числе дефектов.

Коэффициент регрессии:

1 вариант — $\frac{\text{количество дефектов в старом инструментарии}}{\text{количество дефектов в новом инструментарии}}$

2 вариант — $\frac{\text{количество тест-кейсов в статусе failed}}{\text{количество тест-кейсов в регрессионном наборе}}$

Тестировщики следят, сколько дефектов появляется в уже протестированном инструментарии при добавлении новых функций. Если их много, это свидетельствует о низком качестве разработки.



Стоп-фактор — решение о приостановке тестирования. Учитывается текущее значение метрик, выполнение и успешное прохождение тест-кейсов.

Если процент тест-кейсов, выполненных успешно, низкий (какой процент считать низким и достаточным для принятия организационных решений, каждый проект определяет самостоятельно), то тестирование может быть приостановлено до исправления дефектов.

Например, из 100 тест-кейсов, запланированных к исполнению в итерации, выполнено 40. Из них 29 были неуспешными. Менеджер проекта принимает решение о приостановке тестирования до исправления самых критичных дефектов, так как продолжение тестирования приведёт только к нарастанию изъянов, на исправление которых у разработчиков не будет времени.

Есть несколько метрик, которые используются в процессе тестирования:

- степень взаимосвязанности требований;
- коэффициент стабильности требований;
- коэффициент повторно открытых дефектов;
- средняя стоимость исправления дефекта;
- скорость работы команды QA;
- среднее время жизни дефекта;
- коэффициент ошибок, пропущенных в продакшн (в релиз);
- доля неподтверждённых дефектов.

Критерии тестирования

Перед проведением тестирования устанавливаются ориентиры, которые позволяют принимать управленческие решения в процессе работы. Критерии тестирования описываются в плановой документации. А также критерии устанавливаются различными договорённостями между сотрудниками проекта.

Критерии начала тестирования — когда запускать тестирование на проекте. Определение этого критерия при планировании тестирования уменьшает потери времени на заведомо неэффективную работу. Критерии начала тестирования:

- готовность тестовой платформы;
- законченность разработки требуемого инструментария;
- наличие требуемой документации;
- выход новой версии продукта;



- успешный прогон Unit-тестов;
- успешный прогон Smoke-тестов.

Критерии приостановки тестирования — когда остановить тестирование, чтобы ликвидировать проблемы.

1. Критичный баг препятствует тестированию.
2. Накопилось много багов: очевидно, после починки надо всё перепроверять.

Критерии возобновления тестирования — когда возобновить тестирование после приостановки.

1. Исправление критичного дефекта.
2. Выпуск новой, более стабильной, сборки.

Критерии завершения тестирования — когда тестирование считается оконченным.

1. Выполнение более 90% запланированных на итерацию тест-кейсов.
2. Выдержка конкретного периода без открытия новых дефектов.
3. Истечение времени, отведённого на тестирование.
4. Отсутствие дефектов высокой степени важности.

Внесение этого критерия в проектную документацию позволяет избежать преждевременного прекращения тестирования со стороны заказчика.

Приёмочные критерии:

1. Критерии, на основании которых заказчик считает свои требования выполненными.
2. Критерии, которыми будут руководствоваться для вывода продукта на рынок.
 - успешное прохождение 100% тест-кейсов уровня дымового тестирования (smoke-тестов) и 90% тест-кейсов уровня критического пути;
 - условия по устранению 100% дефектов критической и высокой важности;
 - условие, что итоговое покрытие требований тест-кейсами составит не менее 80%.

Использование указанных критериев делает процесс тестирования более прозрачным и снижает риски некачественной разработки или принятия заказчиком неожиданных решений.



Контрольные вопросы

1. Какие разделы в себе содержит отчет о тестировании фичи?
2. Что должен в себе содержать комментарий, который пишется в качестве отчета о тестировании задачи?
3. Что в себе должен содержать отчет об исследовательском тестировании?
4. Как часто формируются отчеты о тестировании?
5. На какие виды делятся метрики?
6. Какие есть критерии тестирования?
7. Приведите пример прямой метрики
8. Приведите пример расчетной метрики

Дополнительные материалы

1. [Как составить отчет о тестировании](#)
2. [Пример отчета о тестировании](#)
3. [Всё, что вам нужно знать о форматах отчётов в тестировании ПО](#)