




Райффайзен
БАНК

Задачи тестировщика

Обучающее видео





Задачи тестировщика

- Работа с требованиями
- Планирование тестирования
- Оценка
- Выполнение тестирования
- Работа с дефектами
- DevOps
- Отчетность



Работа с требованиями





С чего начинается тестирование ПО?

Любое тестирование должно начинаться с анализа требований по задаче.

Требование –

это совокупность утверждений относительно атрибутов, свойств или качеств программной системы, подлежащей реализации.



Характеристики качества требований

Точность

Однозначность (недвусмысленность)

Полнота

Непротиворечивость

Приоритет

Проверяемость

Модифицируемость

Трассируемость



Общее требование

В системе N необходимо заполнять поля значениями по умолчанию.



Вопросы:

- сколько полей?
- где они находятся?
- как они называются?
- какие значения по умолчанию?
- заполнение по умолчанию от чего-то зависит?
- значения по умолчанию можно изменить?



Детальное требование

В системе N на форме создания кредита необходимо заполнять **следующие поля** значениями по умолчанию:

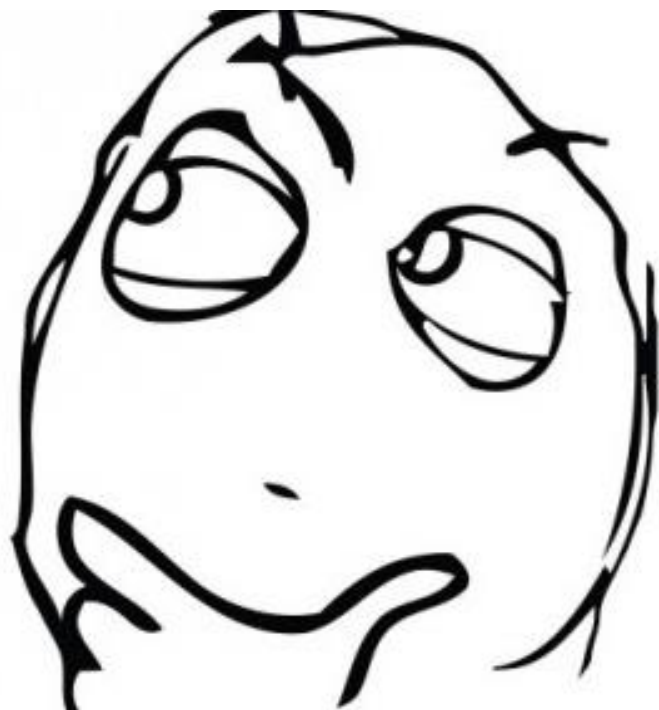
1. Сложная сделка (Complicated deal) = No;
2. Дата начала действия кредита (Start Date) = текущая дата;
3. Дата подписания договора (Deal Date) = текущая дата;
4. Тип выплат (Repayment type) = только основной долг (Principle Only).

Значения по умолчанию должны устанавливаться **при открытии формы** создания кредита, вне зависимости от того, какими значениями заполнены остальные поля на форме.

Перечисленные поля должны остаться доступны для редактирования, все значения по умолчанию могут быть изменены пользователем в процессе заполнения формы.



Стоит ли тратить время на анализ требований?





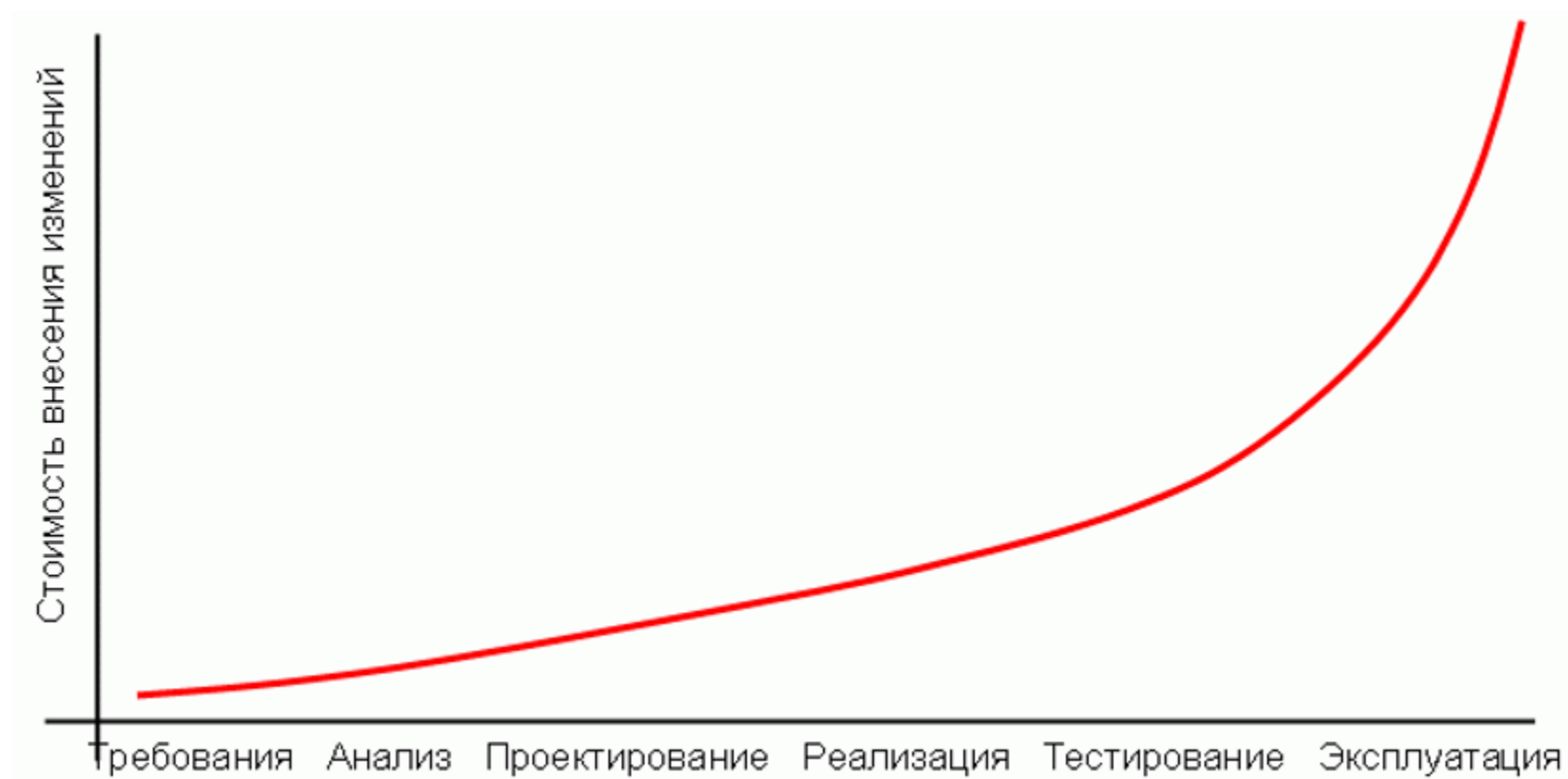
Стоит ли тратить время на анализ требований?

Да, потому что...

- у команды появляется единое представление о задаче
- учитываются риски и особенности реализации задачи
- шанс, что задачу нужно будет переделать, значительно снижается



Цена ошибки





Планирование тестирования





Планирование тестирования

Необходимо определить и описать:

- цели тестирования
- объекты тестирования, виды и уровни тестирования
- критерии начала, приостановки и завершения работ по тестированию
- существующие ограничения тестирования
- этапы проведения работ, сроки и трудозатраты
- необходимые ресурсы
- порядок отчётности



Оценка тестирования





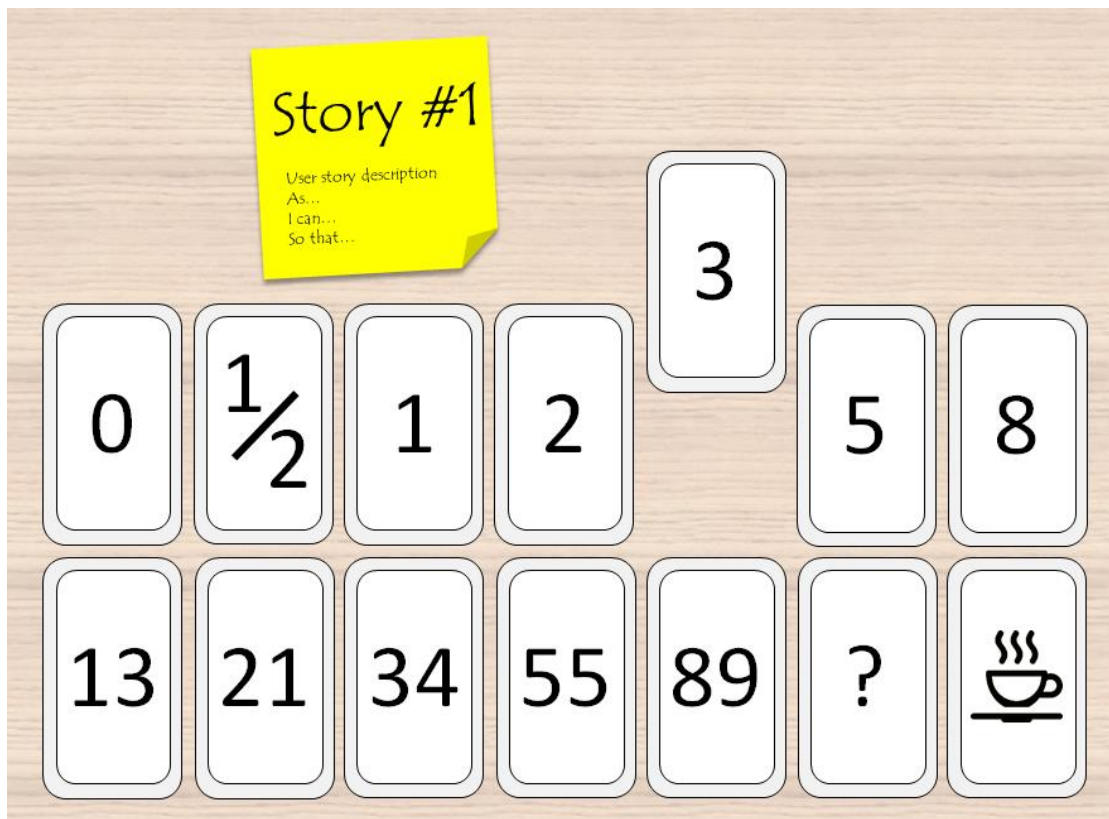
Оценка тестирования - Waterfall

Задача	Трудозатраты на 1 тест	Общие трудозатраты
Составление 40 тестов	0.34 ч.ч.	13.6 ч.ч.
Выполнение 40 тестов	0.5 ч.ч.	20 ч.ч.
Ожидаем около 12ти дефектов (предполагаем, что 30% тестов найдут дефекты)		
Проверка исправления 12 дефектов	0.42 ч.ч.	5 ч.ч.
		46.6 ч.ч.

Итого, трудоемкость **одной итерации** тестирования: **46.6** ч.ч.



Оценка тестирования - Scrum

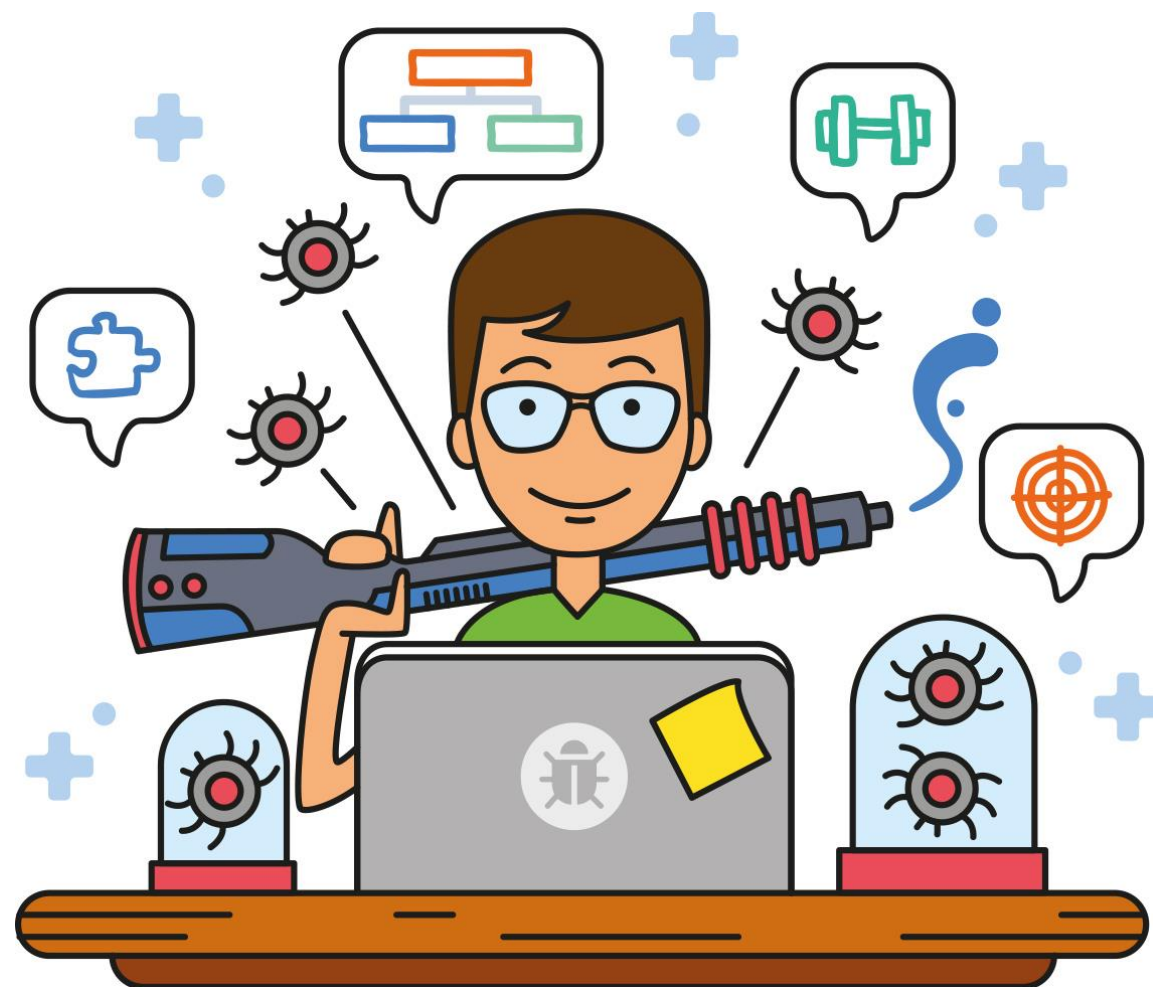


В Scrum **оценка** задач происходит **не отдельно по функциям** аналитики, разработки и тестирования, а **всей задаче целиком**.

Оценка задачи дается **исходя из объема и сложности работы**, без привязки ко времени и рискам.



Выполнение тестирования





В чем заключается выполнение тестирования?

- Ручное выполнение тест-кейсов
- Автоматизированное выполнение тест-кейсов
- Фиксация результатов в Adaptavist



Test Management System - Adaptavist

Test Management for JIRA (Adaptavist) –

стандартный инструмент отдела контроля качества по работе с тестовыми сценариями

Задачи:

- Хранение и управление тест-кейсами
- Сохранение истории выполнения тест-кейсов
- Отчетность по проведенному тестированию
- Интеграция с системами автотестирования Банка



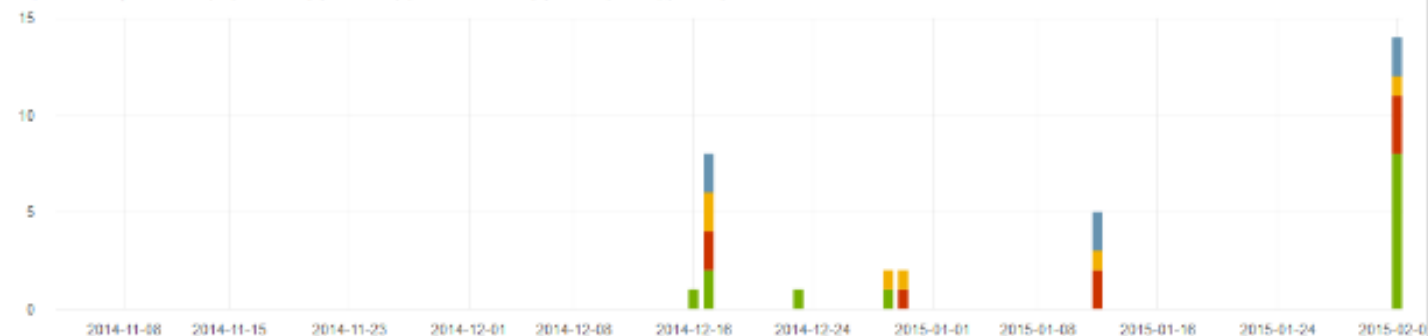
Test Management System - Adaptavist

Test Execution

- Test execution results (summary)
- Test execution results (list)
- Test execution results (detailed)
- Test execution assignment
- Test execution results impacted by issues
- Test execution burn up
- Test execution burn down
- Test execution results over time
- Test execution results over time (completed)
- Test execution results over time (accumulated)
- Test execution effort over time
- Test execution effort (overall)
- Test execution effort by tester
- Test execution results (overall)
- Test execution results (progress)
- Test execution results by project
- Test execution results by coverage
- Test execution results by test plan
- Test execution results by test cycle
- Test execution results by tester
- Test execution results by component
- Test execution results by environment
- Test execution results by label

DAILY TEST EXECUTION PROGRESS

Zoom Out | PASS (13) FAIL (8) WIP (5) BLOCKER (6) count per 1d | (33 hits)



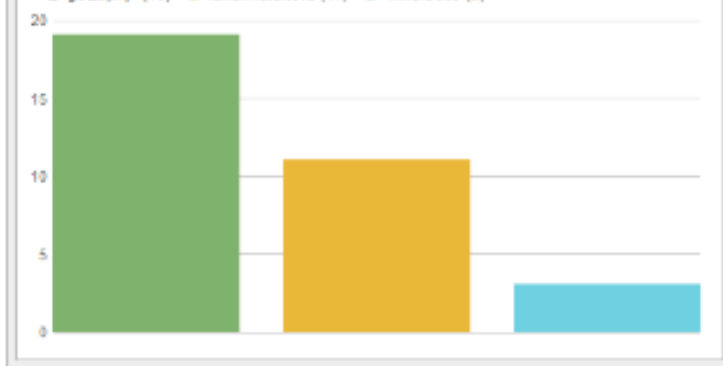
TEST EXECUTIONS BY TEST CYCLE

Security Testing (8) Mobile Testing (8) CLONE - 2 (7) 2 (7)
Regression Mobile Testing (6) Ad hoc (4)



TEST EXECUTIONS BY TESTER

getzephyr (18) lana.malakova (11) mike.betts (3)





Test Management System - Adaptavist

 BPM ▾

Tests

Plans

Cycles

Reports

⋮

REPORTS

- Welcome
- Test Execution (41)
- Test Library (9)
- Traceability (4)
- Issues (17)

[Switch to legacy reports](#)

Test Library > Test cases (summary)

Project (1) clear

BPM ×

Filter by

Folder ▾

clear

Folders

/Test ×

🗕

Generate

Back



Работа с дефектами

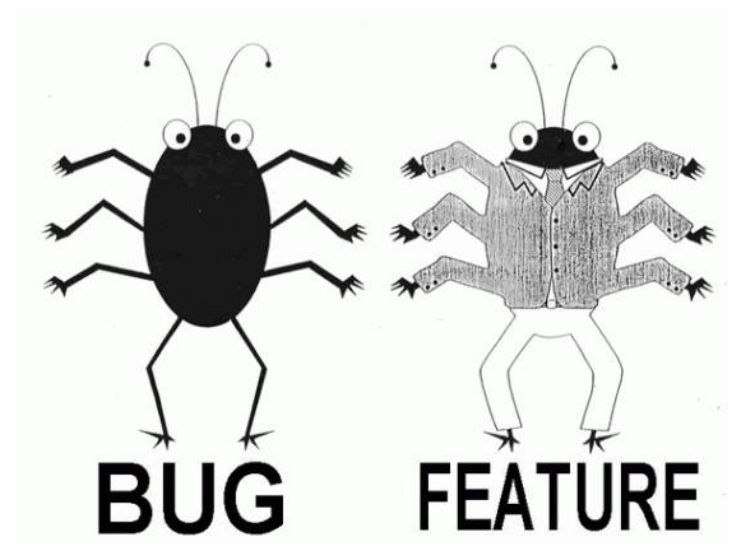




Дефект (баг)

Изыян в компоненте или системе, который может привести компонент или систему к невозможности выполнить требуемую функцию, например неверный оператор или определение данных.

Дефект, обнаруженный во время выполнения, может привести к отказам компонента или системы.





Работа с дефектами

Из чего состоит дефект?

- Уникальный идентификатор
- Уникальный заголовок
- **Шаги для воспроизведения**
- **Ожидаемый результат**
- **Фактический результат**
- Приоритет
- Описание
- Ссылка на тест-кейсы
- Дополнительная информация (логи, скриншоты)



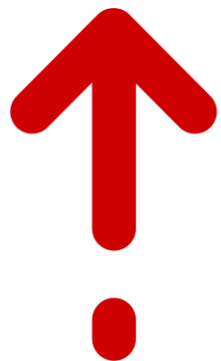
Работа с дефектами

Каким должен быть правильно оформленный дефект?

- Информативный
- Идентифицирующий проблему
- Легко воспроизводимый
- Обладает атомарностью (один дефект - одна проблема)
- Удовлетворяет правилам оформления



Приоритет дефекта (Priority)



Critical



Medium



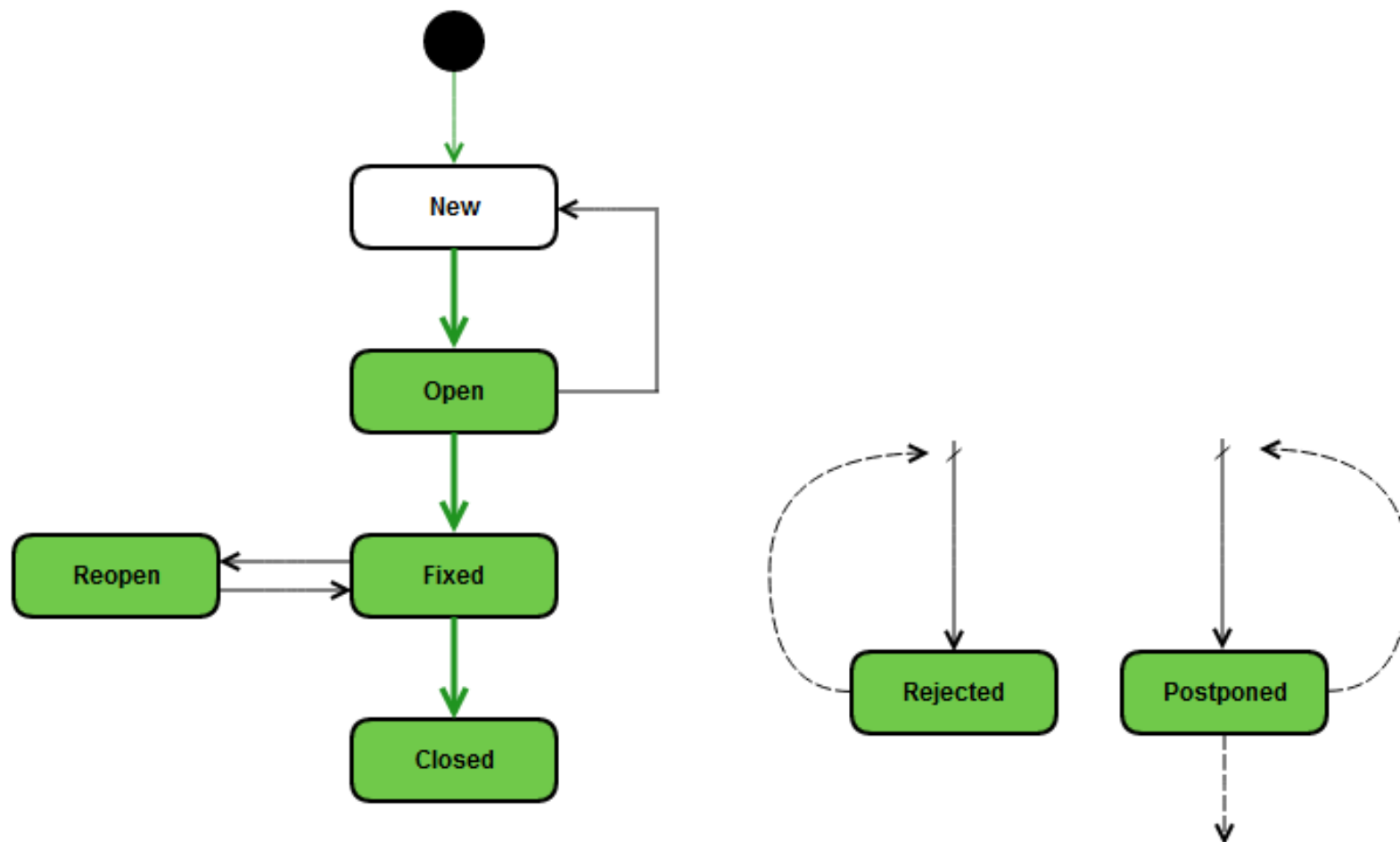
High



Low

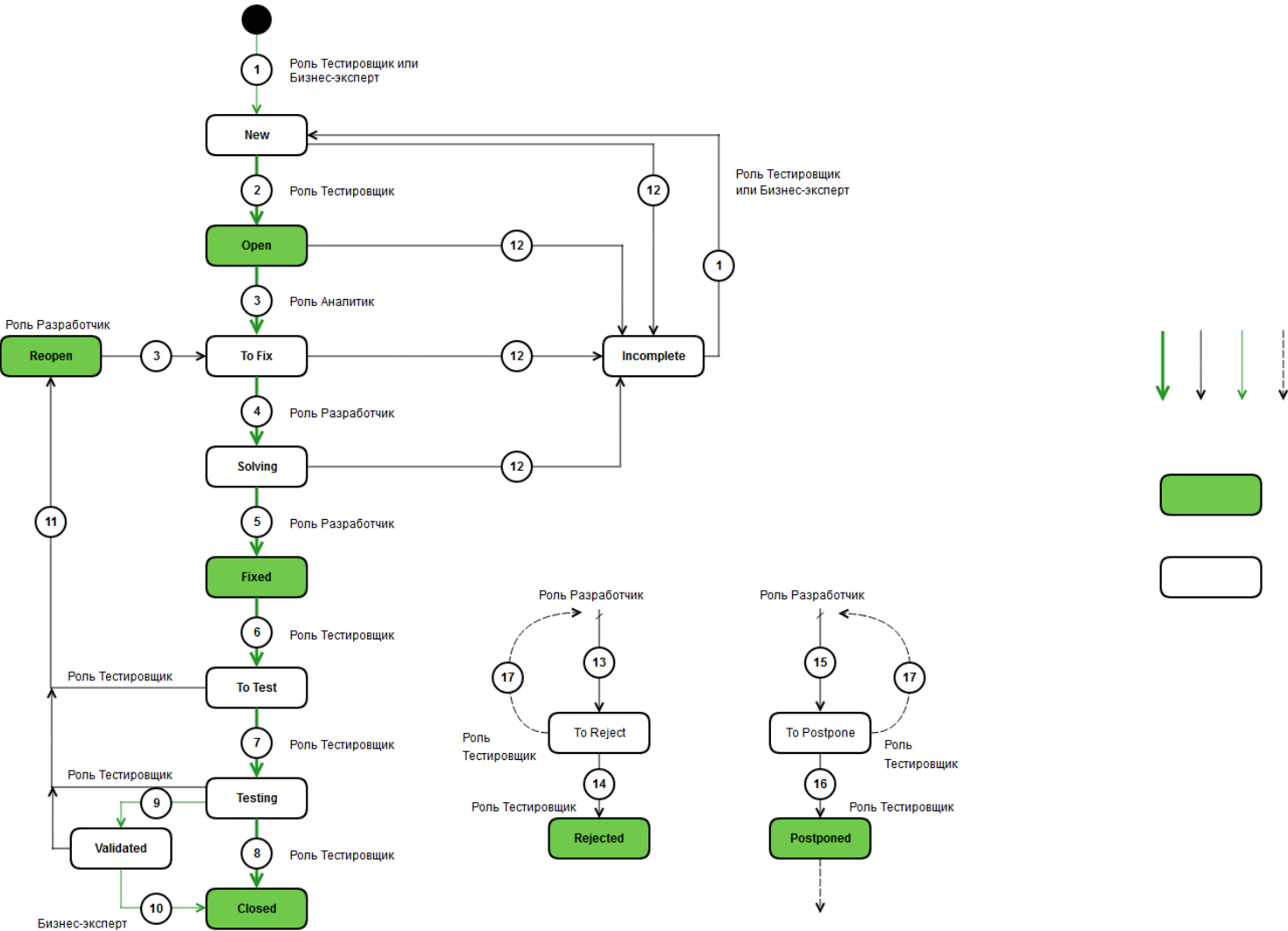


Жизненный цикл дефекта





Жизненный цикл дефекта





Работа с дефектами

Из чего состоит дефект?

- Уникальный идентификатор
- Уникальный заголовок
- Шаги для воспроизведения
- Ожидаемый результат
- Фактический результат
- Приоритет
- Описание
- Ссылка на тест-кейсы
- Дополнительная информация (логи, скриншоты)



Работа с дефектами

Каким должен быть правильно оформленный дефект?

- Информативный
- Идентифицирующий проблему
- Легко воспроизводимый
- Обладает атомарностью (один дефект - одна проблема)
- Удовлетворяет правилам оформления



Работа с дефектами – пример оформления

Плохо оформленный дефект

RAIF-1541. Ошибка трекера

Шаги для воспроизведения: Надеть трекер, походить

Ожидаемый результат: Трекер показывает 20 шагов

Фактический результат: Трекер показывает 1500 шагов

Хорошо оформленный дефект

RAIF-1541. Шагомер. Маленькие расстояния. Ошибка подсчета шагов.

Приоритет: High

Описание: В приложении «Шагомер» присутствует серьезная погрешность в подсчета пройденных шагов. 1 шаг учитывается как 75 шагов. При этом, при прохождении от 1000 шагов, отображение информации на шагомере соответствует реальности.

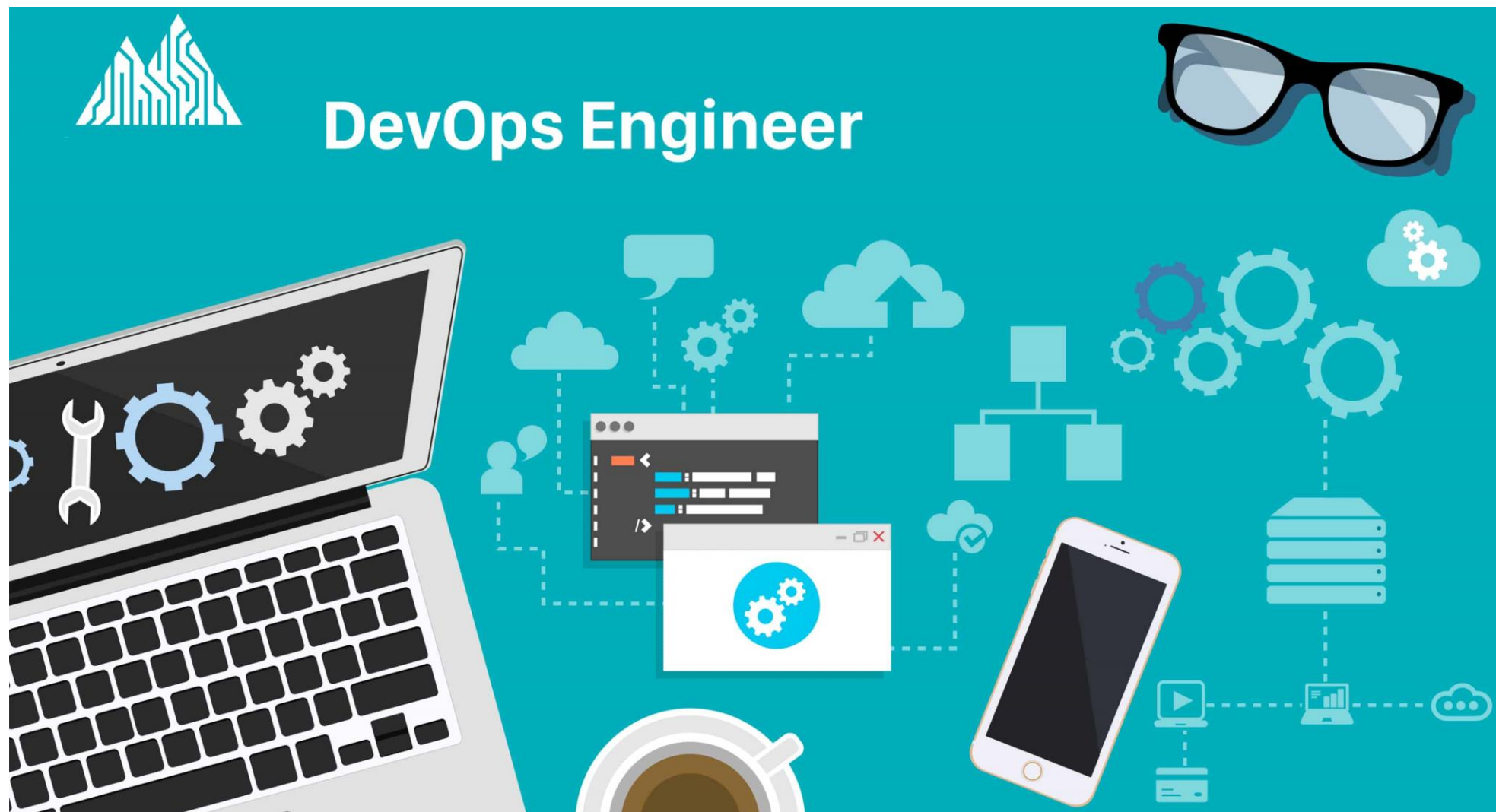
Версия сборки 1.15

Шаги для воспроизведения: Надеть трекер на руку. Сделать 20 шагов.

Ожидаемый результат: Трекер отображает информацию, что пройдено 20 шагов.

Фактический результат: Трекер отображает информацию, что пройдено 1500 шагов.

Работа с тестовыми средами (DevOps)





В чем заключается работа со средами?

- Установка обновлений
- Конфигурация среды
- Установка новых сборок
- Обновление данных



Типы тестовых сред

- Среда тестирования системы
- Интеграционный контур
- Среда нагрузочного тестирования
- Превью (Прод-лайк, Preview)



Отчетность о тестировании





Отчетность о тестировании

Цель составления отчетности –

дать точное представление о состоянии протестированного функционала и обнаруженных дефектах.

Желательно иметь возможность в любой момент времени **автоматически** сгенерировать отчет с нужной информацией.

Для кого формируем отчет?

- Product Owner / Tech Lead
- Команда
- Остальные Stakeholders



Пункты отчета тестирования

- Описание функциональности
- Описание ограничений
- Описание результатов:
 - Состояние тест-кейсов
 - Состояние дефектов
 - Динамика выполнения
- Отклонения от плана тестирования:
 - По работам
 - По срокам
 - По трудозатратам