

< Teach  
Me  
Skills />

Результаты тестирования.  
Дефекты, отчеты



# Поиск ошибок



1

## Основная задача

- завести как можно больше багов

2

## Области тестирования

- самые нестабильные = менее приоритетные для пользователя

3

## Тесты

- нестандартные (негативные, стресс-тесты)

# Тестирование



1

## Основная задача

- пропустить как можно меньше приоритетных для пользователя проблем

2

## Области тестирования

- самые приоритетные для пользователя, даже если они стабильны

3

## Тесты

- стандартные (функциональные)

# Дефект

Баг (bug) – это отклонение фактического результата (actual result) от ожидаемого результата (expected result).

Дефект – поведение программы, затрудняющее (делающее невозможным) пользователю достигнуть цели или удовлетворить интересы. Подразумевает возможность исправления . При невозможности переходит в разряд ограничения технологии

# Дефект

**Ожидаемый результат** - поведение системы, описанное в требованиях. **Фактический результат** - поведение системы, наблюдаемое в процессе тестирования.

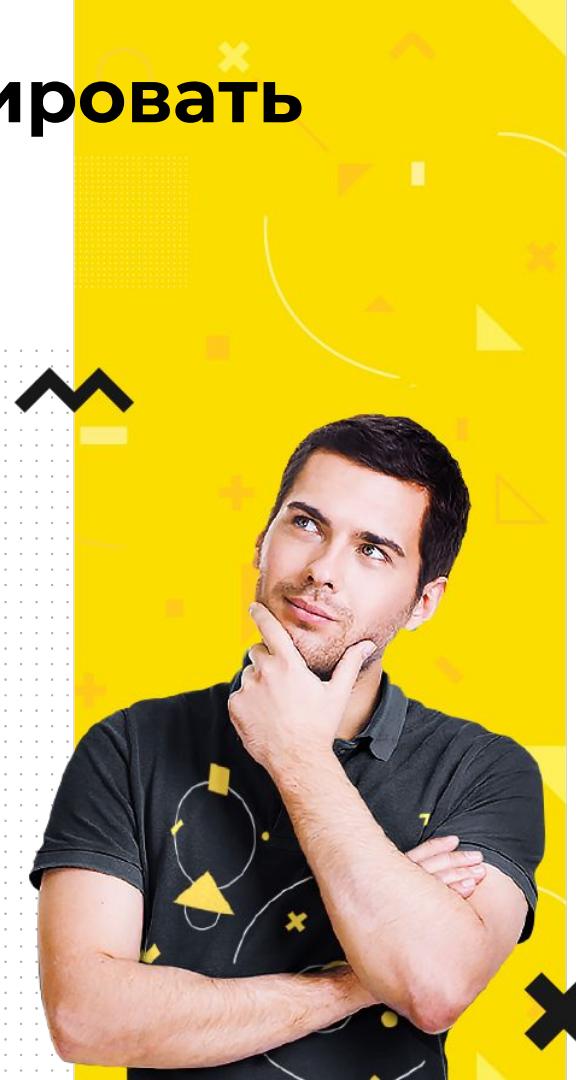
Помните, что к багам относится любое некорректное поведение программы, не соответствующее ожиданиям пользователя, даже в том случае, если оно не описано в требованиях и спецификациях. [Важное тематическое видео](#)

Баги могут встречаться в любой документации, в архитектуре и дизайне, в коде программы...

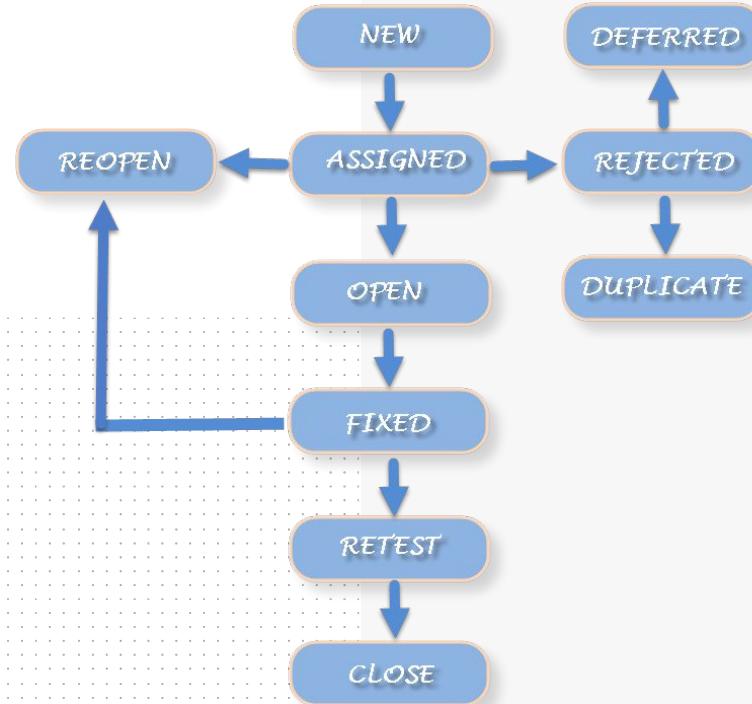
Иногда баг является не ошибкой в программе, а результатом неверного конфигурирования программы и/или окружения.

# Кто может задокументировать баг?

- Тестировщики и специалисты по качеству
- Разработчики
- Представители службы технической поддержки
- Продавцы и специалисты по маркетингу
- Представители заказчика
- Конечные пользователи



# Жизненный цикл дефекта





## Атрибуты дефекта

- Идентификатор (id)
- Краткое описание (summary)
- Подробное описание (description) :
  - Шаги воспроизведения (steps to reproduce, STR)
  - Актуальный результат (actual Result)
  - Ожидаемый результат (expected Result)
- Важность (severity)
- Срочность (priority)

# Идентификатор (Id)

У каждого отчета об ошибке должен быть уникальный идентификатор . Как правило, системы управления ошибками (bug tracking systems) позволяют формировать идентификатор в виде некоторого шаблона

Например : Аббревиатура проекта + дата + порядковый номер WSVELC20080625001 или WS\_VELC\_20080625\_001 или WSVELC - 001

# Краткое описание (summary)

Хорошее краткое описание должно давать ответы на три вопроса:

- Что?
- Где?
- При каких условиях?

Например:

- Нет логотипа на странице приветствия, если пользователь администратор
- Невозможно открыть файл с именем > 500 символов
- Приложение виснет при попытке ввести пустой пароль на странице авторизации

# Шаги воспроизведения (steps)

Описывайте каждый шаг, пока не столкнетесь с дефектом

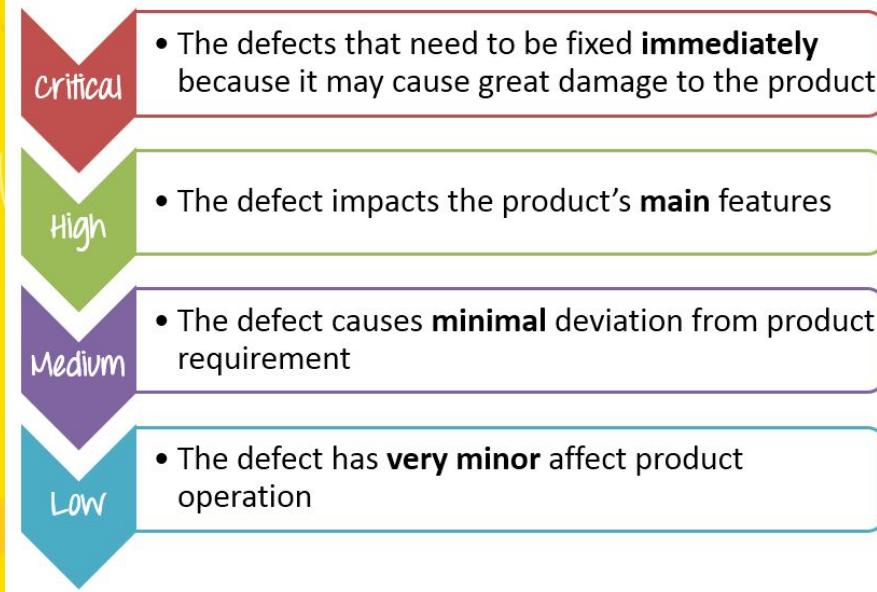
- Найдите точный путь, чтобы воспроизвести дефект
- Попытайтесь найти кратчайший путь
- Повторите все шаги несколько раз и убедитесь, что всё верно
- Описывайте каждое действие в отдельном шаге

# Шаги воспроизведения (пример)

1. Перейти по ссылке: <http://www.site.com/login/>
  2. Ввести в Логин "admin"
  3. Ввести в Пароль "dmpwd"
  4. Кликнуть Войти
- 
- **Actual result:** В левом верхнем углу вместо логотипа – пустое место
  - **Expected result:** В верхнем углу есть логотип

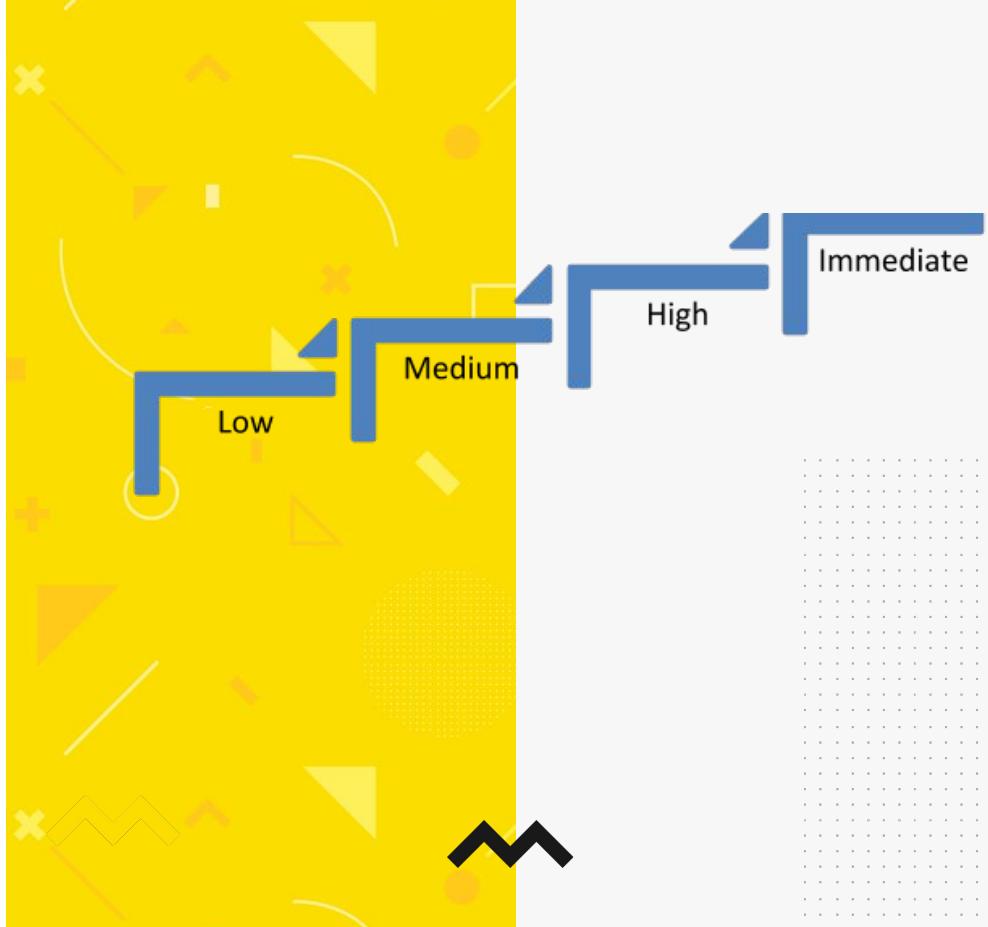


# Severity (важность)



Это поле показывает насколько  
ошибка затрагивает **значимый**  
**функционал приложения**

# Priority (срочность)



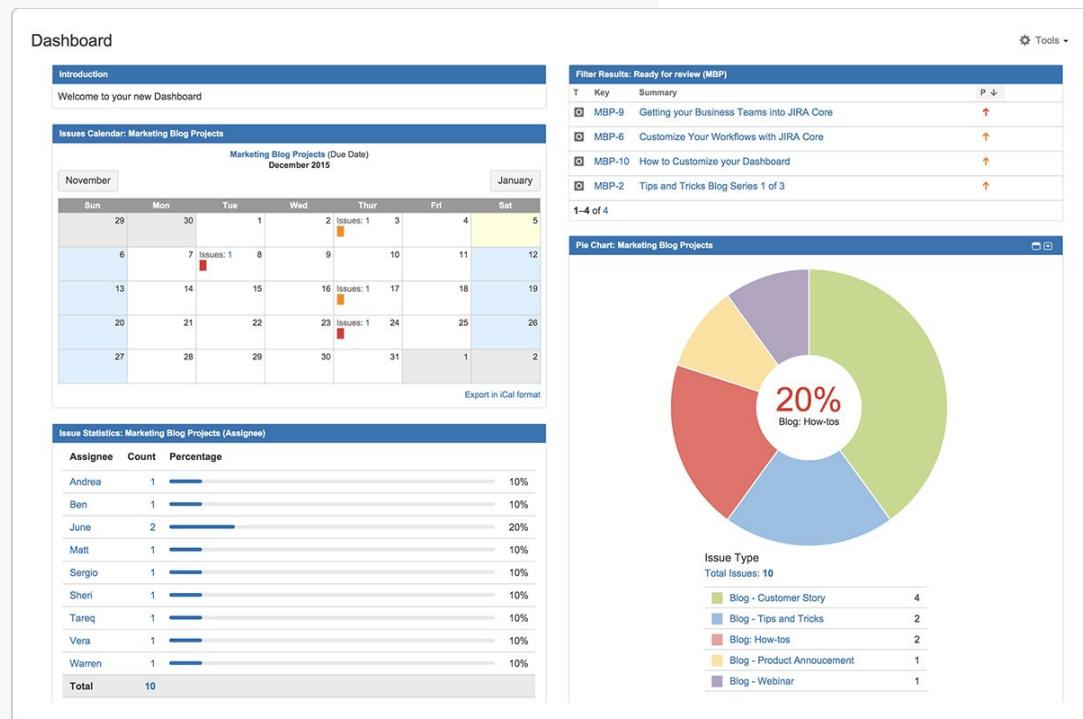
Это поле показывает как быстро  
необходимо исправить ошибку

# Отчет о результатах тестирования

Обычно, TRR предоставляется для ознакомления всей проектной команде и заказчику. Опытный менеджер проектов всегда собирает максимально полную информацию о состоянии дел, прежде чем сделать выводы. Он не станет опираться только на информацию, предоставленную отделом разработчиков, т.к. разработчики часто склонны давать более оптимистичную оценку проекта, чем есть на самом деле. В то же время тестировщики, во-первых, не заинтересованы в приукрашивании отчётов, а во-вторых – обладают более полной информацией о текущем состоянии качества продукта, чем какая бы то ни было другая часть проектной команды.

# Статистика по ошибкам

Сводная таблица со статистикой по найденным ошибкам по типу/периоду времени



# Список дефектов

Как правило выгрузка из баг-трекинговой системы с полным списком проблем

project IN (BM, AB) AND issuetype = Bug and assignee=currentuser() and priority in (blocker, critical) and status in ("To Do", open)

1-9 of 9

T	Key	Summary	Assignee	Reporter	P	Status	Resolution	Created	Updated
	BM-132	Test	Swati Seela [Administrator]	Swati Seela [Administrator]	0	OPEN	Unresolved	17/Feb/16	03/Mar/16
	AB-38	Rest_validatAccount_creditcard_ ; credit card validation gives incorrect error message when Expiry date is prior than the present date.	Swati Seela [Administrator]	Test	↑	TO DO	Unresolved	13/Aug/15	19/Aug/15
	AB-37	Update functionality for SSN.accnts input'0'with success,but does not update in DB	Swati Seela [Administrator]	Test	↑	TO DO	Unresolved	13/Aug/15	19/Aug/15
	AB-35	Timeout Error for server socket	Swati Seela [Administrator]	Test	↑	TO DO	Unresolved	13/Aug/15	19/Aug/15
	AB-29	Buy credit card accepts negative amount and it's adding in available limit in database	Swati Seela [Administrator]	Test	↑	TO DO	Unresolved	13/Aug/15	19/Aug/15
	AB-27	In CC card restful service "Add account" Credit limit ,available limit and cash limit accepts 0 and show "msg". "Successfully saved credit card information".	Swati Seela [Administrator]	Test	↑	TO DO	Unresolved	13/Aug/15	19/Aug/15
	AB-26	In CC restful service "add account" CC expiry date should be greater than the current date but accepts the date prior to the current date.	Swati Seela [Administrator]	Test	↑	TO DO	Unresolved	13/Aug/15	19/Aug/15
	AB-19	If we give non matching name and credit card number i.e name of one person and registered credit card of other person the amount in the request get deposited in the account of name provided.	Swati Seela [Administrator]	Test	↑	TO DO	Unresolved	13/Aug/15	19/Aug/15

## Статистика по автоматизированным тестам



- Количество автотестов
- Соотношение автотестов к ручным тестам
- Общее время запуска
- Количество flaky-тестов (хрупких)
- Время, затрачиваемое на ручные тесты и, соответственно, его уменьшение из-за автоматизации (ROI)