

## QA Theory. Лекция 6

- Проведение тестирования
- Работа с дефектами

# Основы QA процесса

## SDLC

Анализ

Дизайн

Кодирование

Тестирование

Внедрение

## STLC

Планирование

Тест анализ  
*Что тестируем?*

Тест дизайн  
Разработка тестов  
*Как тестируем?*

Выполнение тестов

Анализ и отчет о  
результатах  
тестирования

## Мониторинг и контроль –

непрерывное сравнивание актуального прогресса с планом тестирования

## **Подготовка к началу тестирования (Test environment setup)**

- Расстановка приоритетов активностей выполнения тестирования;
- Создание наборов тест-кейсов (тест-сьютов) для последующего выполнения;
- Создание автоматизированных сценариев;
- Настройка необходимого тестового окружения;
- Подготовка тестовых данных.

*К началу выполнения тестирования должно быть готово и настроено:*

- Тестовое окружение (Test environment);
- Инструменты управления тестированием (Test management system);
- Созданы тест раны (test run) — наборы тестов, которые назначены на определённых людей для прохождения;
- Инструменты управления дефектами (Task and bug tracking systems).
- Билд должен быть собран, юнит тесты прошли, билд развернут
- Весь запланированный новый функционал должен быть реализован

## **TEST EXECUTION**

- Выполнение тестов из выбранных наборов (manual/automation);
- Сравнение фактических и ожидаемых результатов;
- Проставление результатов выполнения тестов (пройден, не пройден, заблокирован и т.д.);
- В случае нахождения и исправления дефекта - Retest (+ регрессионное тестирование).
- Составление отчётов о дефектах;

## **Анализ и отчет о результатах тестирования**

Анализ критериев окончания тестирования и составление отчетов является финальным аккордом процесса тестирования. Именно тут мы понимаем, что протестировали достаточно и можем отдавать продукт в релиз.

### **возможные Критерии окончания тестирования:**

- Устойчивая работа системы
- Время и другие ресурсы исчерпаны
- Все запланированные тесты пройдены без ошибок
- Все найденные ошибки исправлены
- Regression test прошел без ошибок

## Действия при обнаружении ошибки

- **Скриншот!**
- Проверить корректность версии приложения (билда)
- Проверить тестовые данные – убедиться, что ошибка произошла не из-за ввода неверного значения
- Проверить настройки, документацию
- Описать подробно последовательность действий, которые привели к ошибке. Проверить ошибку на других тестовых данных
- Собрать любую полезную информацию: логи, данные в БД, воспроизводимость на других данных
- Перед созданием дефекта по возможности обсудить с разработчиком
- В спорных случаях уточнить у тестировщиков (более опытных) и/или аналитиков

**Дефект** (он же **bug**) – это несоответствие фактического результата выполнения программы ожидаемому результату

Типы дефектов

## Bug. Bug report.

### Минимальные данные о дефекте:

- **ID** – уникальный номер дефекта (*генерируется автоматически*)
- **Summary** - краткие данные
- **Date** - дата
- **Author** - автор (тестировщик)
- **Version** - версия сборки (build)
- **Severity** – как правило заполняет тестировщик (Trivial, Minor, Major, Critical, Blocker)
- **Priority** – как правило заполняет менеджер (Low, Medium, High)
- **Steps to Reproduce** – шаги для воспроизведения (предусловия, ожидаемый результат, фактический результат)
- **Actual results** -
- **Excepted results** -
- **Status** - в зависимости от используемого жизненного цикла – Bug flow and life cycle.
- **Assigned to** – разработчик, назначенные на исправление
- **Attachments**

**LOW**

**Priority**

**HIGH**

**LOW**

**Severity**

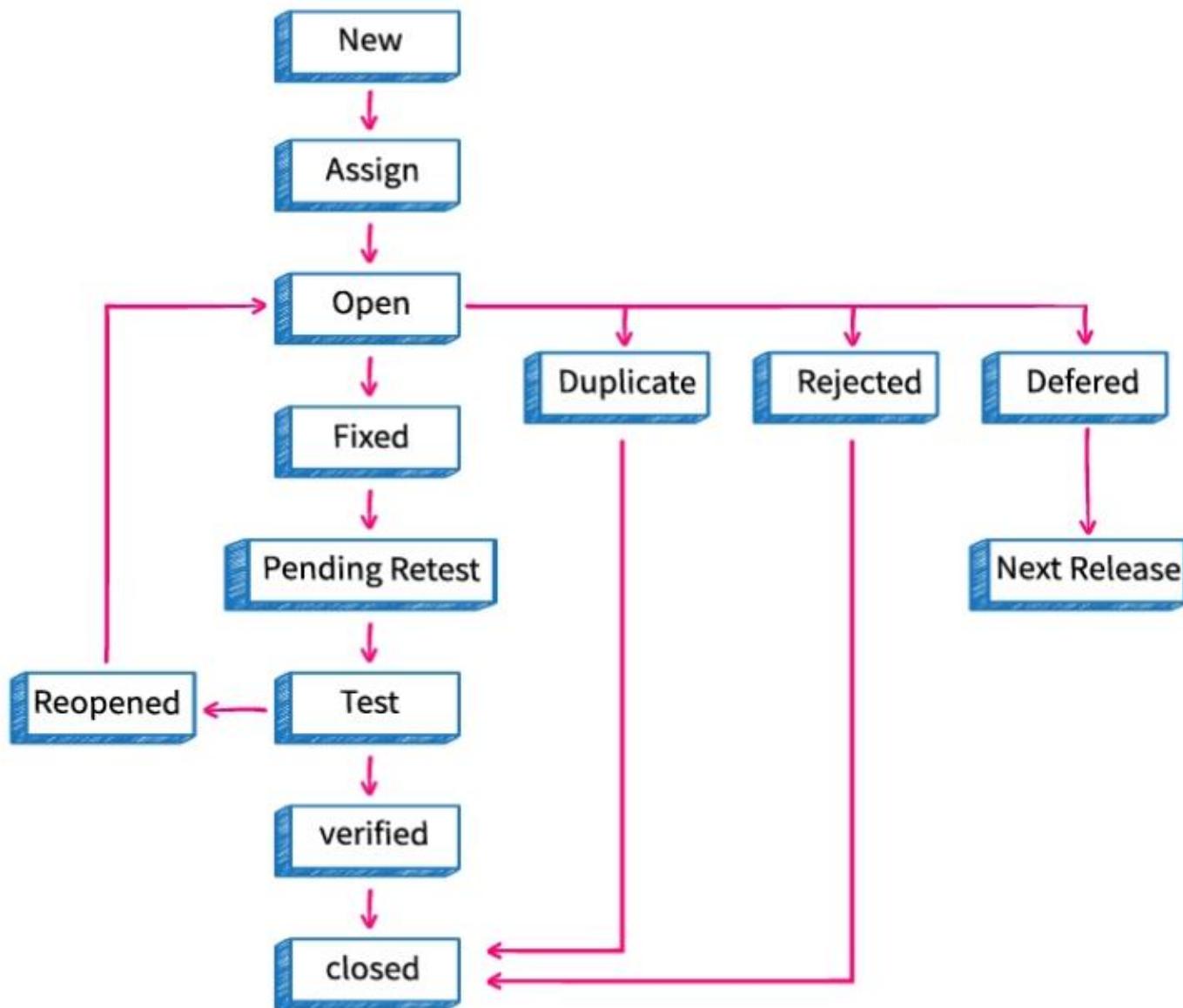
**HIGH**

	<ul style="list-style-type: none"><li>«Условия использования» amazon.com - в тексте есть проблема с выравниванием или орфографические ошибки</li><li>Подпись к картинке написана неправильным шрифтом</li></ul>	<ul style="list-style-type: none"><li>Не работает функция, которая очень редко используется</li><li>Не работает «Уведомление о конфиденциальности» в нижней части главной страницы</li></ul>
<b>Priority</b>	<ul style="list-style-type: none"><li>Цвет логотипа компании не корректный</li><li>amazon.com логотип отображается как «amazn.com» с пропущенной буквой «о».</li><li>на сайте написано «porn content, а надо - «adult content»</li></ul>	<ul style="list-style-type: none"><li>Не работает основной функционал</li><li>Кнопка входа в приложение не работает</li><li>Товары в корзине - кнопку «Перейти к оформлению заказа» дает сбой</li></ul>
<b>HIGH</b>		

**Severity** (серьезность) — степень негативного влияния дефекта на продукт. Выставляет тестировщик, показывает влияние дефекта на работоспособность приложения.

**Priority** (приоритет) — порядок в котором дефекты должны быть исправлены. Определяются разработкой и бизнесом (выставляют программисты, РМ, TeamLead проекта). Чем выше стоит приоритет, тем скорее нужно исправить дефект

# Bug/Defect Life Cycle



## **Defect examples**

Jira - example 1

Jira - example 2

# Defect example – Azure DevOps

State	Closed	Area	Virtual_Events
Reason	Moved out of state QA	Iteration	Virtual_Events\2023_Q3\Release 1.2.4

## Repro Steps

Planning

Description: When refunding a user with the correct payment method we are unable to receive a correct refund message.

Platform: PC / Windows 11



Effort (Hours)

Browser: Chrome, Edge, Firefox

Original Estimate

Environment: Dev env.

Remaining

Version: 1.1.8

Completed

Repro attempts: 5/5

Labels: Payments

## Steps to reproduce:

1. Go to the event, Key 400
2. Set up payment with discount selected
3. Using a Payment page complete payment
4. Go to registration Activity, select the same user
5. Click on the Payment tabs to complete Refund or this user
6. Observe an issue

Actual result: Refund not completed. Incorrect message displayed

Expected result: Refund should be completed successfully

Attachments: Screenshots, Videos

Linked issues: Blockers / Duplicates / Test Cases / Story

# Defect example – Azure DevOps

Virtual\_Events\2023\_Q3\Release 1.2.4

Details 1 2

Planning Deployment

Effort (Hours)

Original Estimate

Remaining

Completed

To track releases associated with this work item, go to [Releases](#) and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting

Development

Related Work

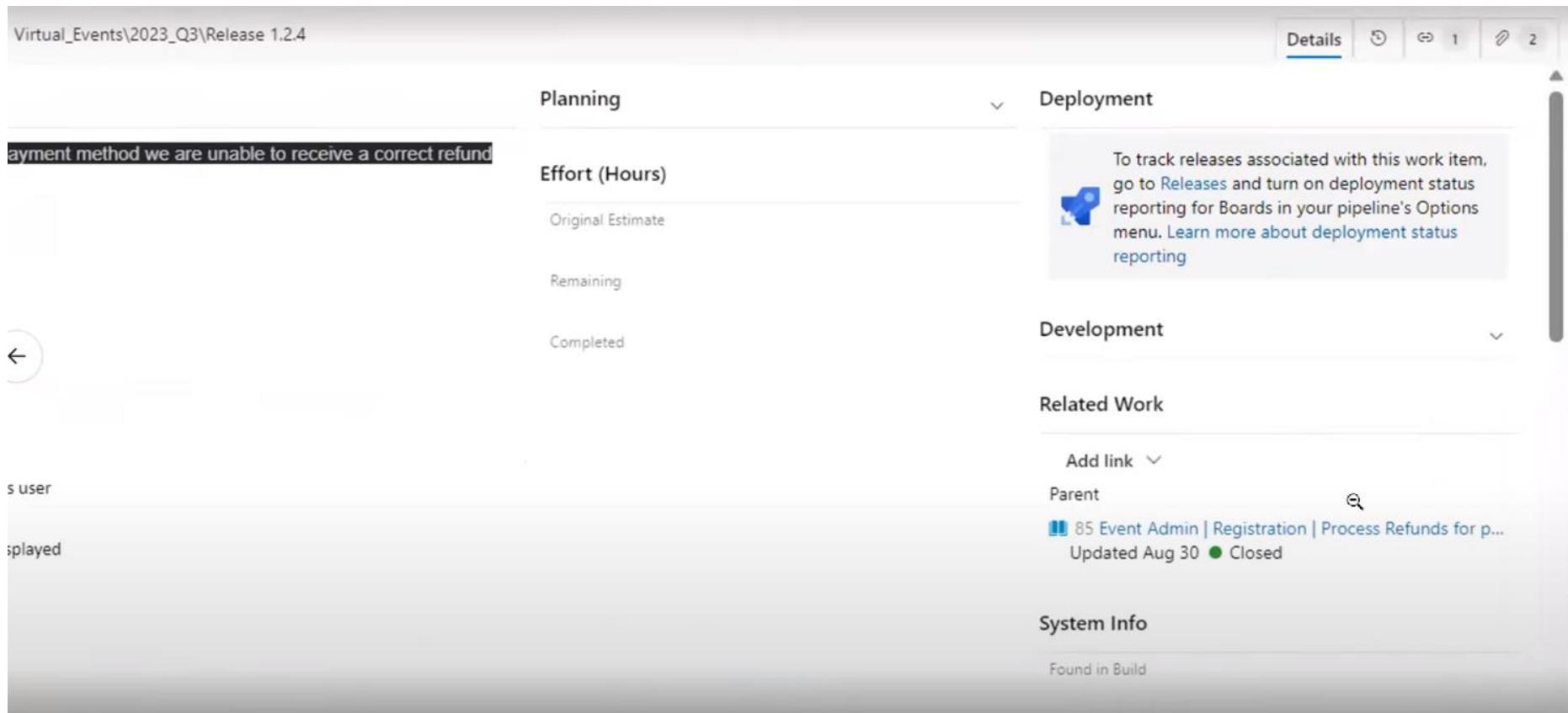
Add link ▾

Parent

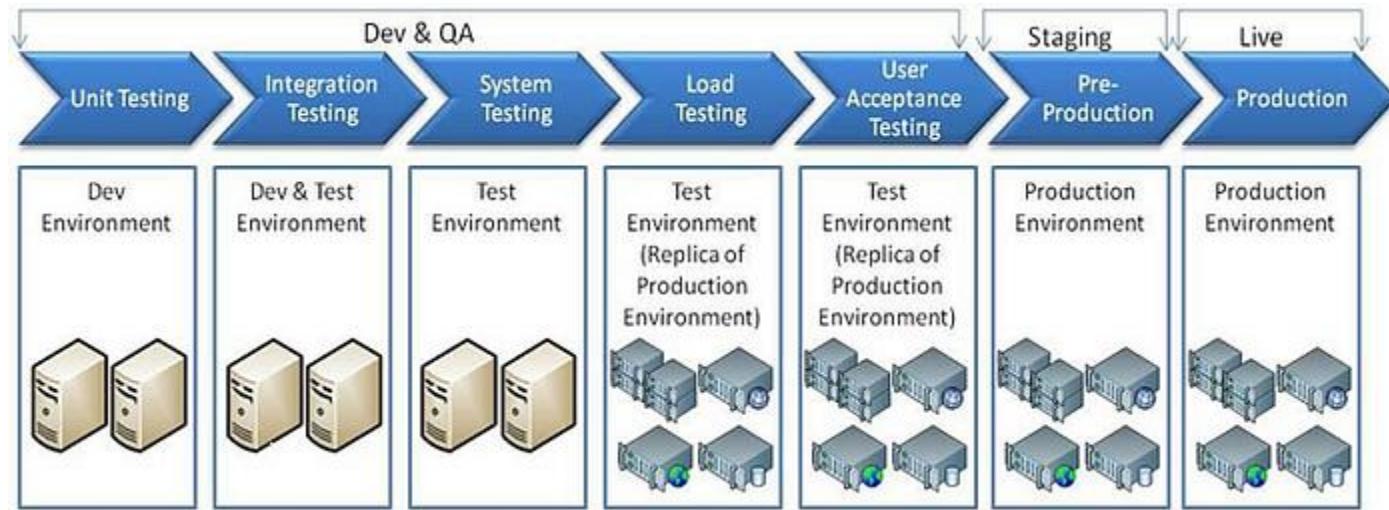
85 Event Admin | Registration | Process Refunds for p...  
Updated Aug 30 • Closed

System Info

Found in Build



+ attachments



## **Environments** - Окружения разёртывания программного обеспечения

### **Local**

Компьютер разработчика

### **Development/ DEV**

Сервер разработки выступающий как песочница где разработчик может выполнить unit-тестирование

### **Testing/Test/QA/Internal Acceptance/ACP/ SYS**

Окружение в котором выполняется тестирование интерфейса или тестирования без связи с внешними системами

### **Integration / INT**

Основа для интеграционного тестирования или для построения CI

### **Staging/Stage/Model/Pre-production/External-Client Acceptance/Demo/HotFix**

**REL / PVS** – performance volume stress/ **PFIX** – prod fix

Зеркало продакшн-окружения

### **Production/Live/ PROD**

Серверы конечных пользователей/клиентов

## Тренировочные сайты

<http://testingchallenges.thetestingmap.org/index.php>

<https://demo.prestashop.com/#/en/front>

## Примеры дефектов Accessibility testing

<https://alphagov.github.io/accessibility-tool-audit/test-cases.html#content>