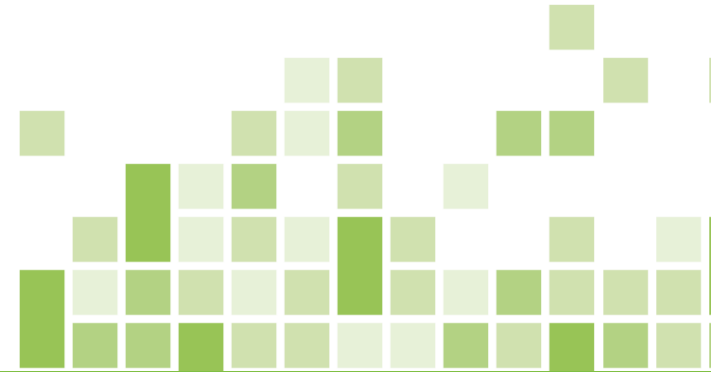




ТОМСКИЙ  
ПОЛИТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ



# Тестирование программного обеспечения

## Лекция №8.

### Отчет о результатах тестирования

#### Метрики

Копнов Максим Валериевич

Томск  
2021

# Вспоминаем прошлое занятие

---

- **Дефекты**
  - Отчёт о дефектах – жизненный цикл
  - Отчёт о дефектах – атрибуты

# О чём сегодня будем говорить

---

- **Дефекты**
  - Свойства качественных отчётов о дефектах
- **Метрики тестирования**
  - Определение
  - Цели
  - Виды
  - Общепринятые метрики
  - Свойства
  - Метрики покрытия
- **Автоматизация тестирования**

# Свойства качественных отчётов о дефектах

---

- Тщательное заполнение всех полей точной и корректной информацией.
- Правильный технический язык.
- Специфичность описания шагов.
- Отсутствие лишних действий и/или их длинных описаний.
- Отсутствие дубликатов.
- Очевидность и понятность.
- Прослеживаемость.
- Отдельные отчёты для каждого нового дефекта.
- Соответствие принятым шаблонам оформления и традициям.

# Метрики – определение

---

**Метрика** — числовая характеристика показателя качества. Может включать описание способов оценки и анализа результата.

# Метрики – цели

---

- Принятие решений о начале, приостановке, возобновлении или прекращении тестирования определения степени соответствия продукта заявленным критериям качества.
- Определение степени отклонения фактического развития проекта от плана.
- Выявление «узких мест», потенциальных рисков и иных проблем.
- Оценка результативности принятых управленческих решений
- Подготовки объективной информативной отчётности.

# Метрики – виды

---

- Прямые – не требуют вычислений.
  - Общее количество выполненных тесткейсов
  - Количество успешно пройденных тесткейсов
  - Количество «проваленных» тесткейсов
  - ...
- Расчётные – вычисляются по формуле.

# Метрики – виды

---

- Простая расчётная метрика.

$$T^{SP} = \frac{T^{Success}}{T^{Total}} \cdot 100\%$$

$T^{SP}$  — процентный показатель успешного прохождения тесткейсов

$T^{Success}$  — количество успешно выполненных тесткейсов

$T^{Total}$  — общее количество выполненных тесткейсов

Минимальные границы значений:

- Начальная фаза проекта: 10 %.
- Основная фаза проекта: 40 %.
- Финальная фаза проекта: 85 %



# Метрики – виды

---

- Сложная расчётная метрика .

$$T^{SC} = \sum_{Level}^{MaxLevel} \frac{(T_{Level} \cdot I)^{R_{Level}}}{B_{Level}}$$

$T^{SC}$  – интегральная метрика прохождения тест-кейсов во взаимосвязи с требованиями и дефектами.

$T_{Level}$  – степень важности тесткейса.

$I$  – количество выполнений тесткейса.

$R_{Level}$  – степень важности требования, проверяемого тесткейсом.

$B_{Level}$  – количество дефектов, обнаруженных тесткейсом.

- Идеальным состоянием является непрерывный рост значения  $T^{SC}$ .
- В случае отрицательной динамики уменьшение значения  $T^{SC}$  на 15 % и более за последние три спринта может трактоваться как недопустимое и являться достаточным поводом для приостановки тестирования

# Общепринятые метрики

---

- Процентное отношение (не) выполненных тест-кейсов ко всем имеющимся.
- Процентный показатель успешного прохождения тест-кейсов.
- Процентный показатель заблокированных тест-кейсов.
- Плотность распределения дефектов.
- Эффективность устранения дефектов.
- Распределение дефектов по важности и срочности.
- ...

# Метрики – свойства

---

- Основная цель любой метрики — это улучшение процесса разработки и самого программного продукта. Метрика позволяет увидеть, в какой точке на пути к целям мы находимся в данный момент, приближаемся к ним или удаляемся, достигаются ли критерии успешности.
- Метрики не должны существовать ради самого процесса измерения. Необходимо определить зоны для оптимизации, а потом решать, как их оценивать.

# Метрики – покрытие

---

- **Покрытие** — процентное выражение степени, в которой исследуемый элемент затронут соответствующим набором тест-кейсов.

# Метрики – покрытие

---

- Покрытие требований

$$R^{SimpleCoverage} = \frac{R^{Covered}}{R^{Total}} \cdot 100\%, \text{ где}$$

$R^{SimpleCoverage}$  — метрика покрытия требований,

$R^{Covered}$  — количество требований, покрытых хотя бы одним тест-кейсом,

$R^{Total}$  — общее количество требований.

- Плотность покрытия требований

$$R^{DensityCoverage} = \frac{\sum T_i}{T^{Total} \cdot R^{Total}} \cdot 100\%, \text{ где}$$

$R^{DensityCoverage}$  — плотность покрытия требований,

$T_i$  — количество тест-кейсов, покрывающих  $i$ -е требование,

$T^{Total}$  — общее количество тест-кейсов,

$R^{Total}$  — общее количество требований.

# Метрики – покрытие

---

- Покрытие классов эквивалентности

$$E^{Coverage} = \frac{E^{Covered}}{E^{Total}} \cdot 100\%, \text{ где}$$

$E^{Coverage}$  — метрика покрытия классов эквивалентности,

$E^{Covered}$  — количество классов эквивалентности, покрытых хотя бы одним тест-кейсом,

$E^{Total}$  — общее количество классов эквивалентности.

- Покрытие граничных условий

$$B^{Coverage} = \frac{B^{Covered}}{B^{Total}} \cdot 100\%, \text{ где}$$

$B^{Coverage}$  — метрика покрытия граничных условий,

$B^{Covered}$  — количество граничных условий, покрытых хотя бы одним тест-кейсом,

$B^{Total}$  — общее количество граничных условий.

- Покрытие кода модульными тест-кейсами.

# Автоматизация тестирования – определение

---

**Автоматизированное тестирование** – набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования.

# Автоматизация тестирования – преимущества

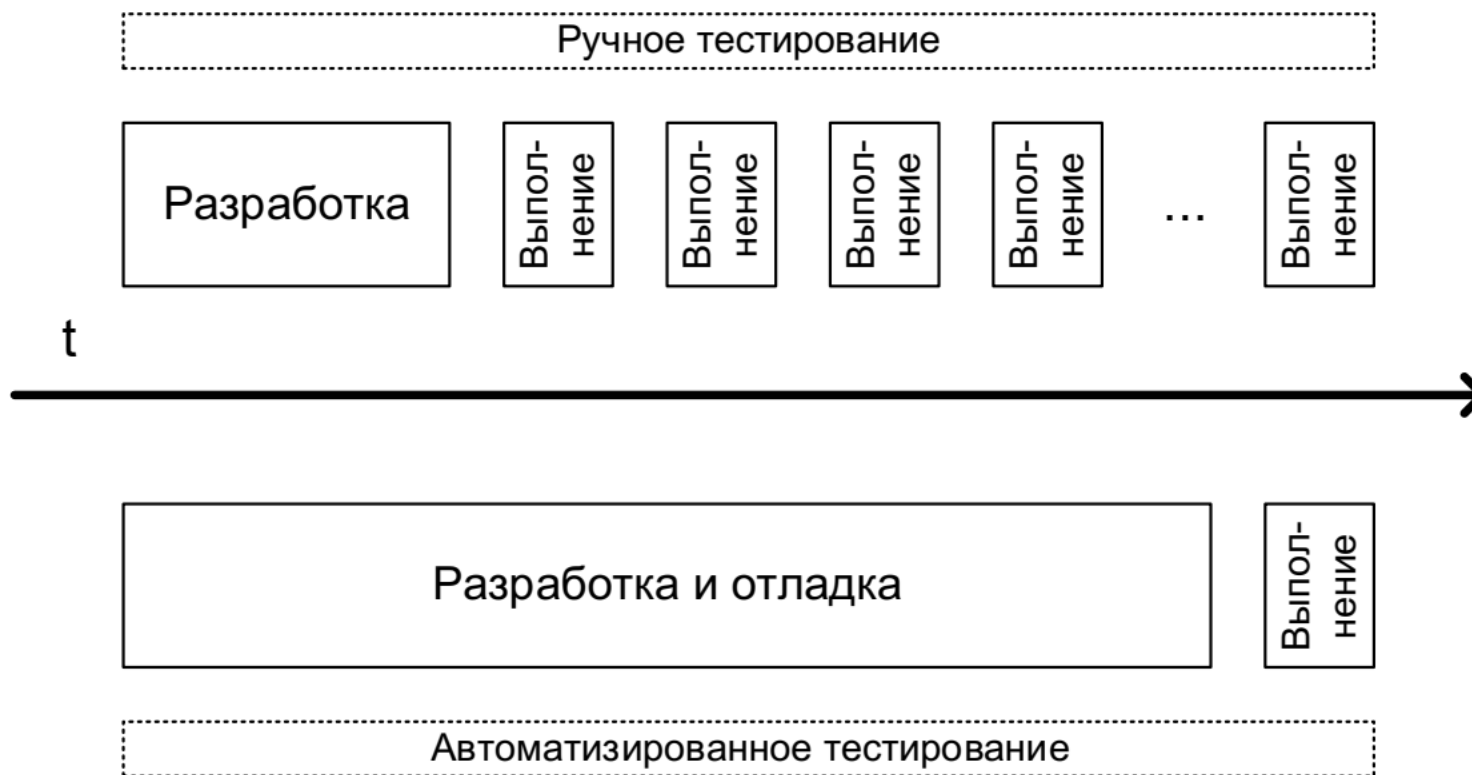
---

- Высокая скорость выполнения тесткейсов.
- Отсутствие влияния человеческого фактора.
- Выполнение тесткейсов, которые не способен выполнить человек.
- Автоматический сбор, хранение, анализ, агрегация и представление исторических данных.



# Автоматизация тестирования – недостатки

- Требуется высококвалифицированный персонал.
- Затраты на разработку и сопровождение тестов.
- Стоимость средств автоматизации.



# Автоматизация тестирования – задачи которые автоматизируют

---

- Регрессионное тестирование.
- Инсталляционное тестирование и настройка тестового окружения.
- Конфигурационное тестирование и тестирование совместимости.
- Использование комбинаторных техник тестирования.
- Модульное тестирование.
- Интеграционное тестирование.
- Тестирование производительности.
- Дымовое тестирование.
- Приложения без графического интерфейса.
- Рутинные, или требующие повышенного внимания операции.
- Проверка «внутренней функциональности» веб-приложений.

# Автоматизация тестирования – задачи которые не автоматизируют

---

- Функциональность, которую нужно (достаточно) проверить всего несколько раз..
- Тест-кейсы, которые нужно выполнить всего несколько раз.
- Низкий уровень абстракции в имеющихся инструментах автоматизации.
- Низкая стабильность требований.
- Проблемы с планированием и ручным тестированием.
- Области тестирования, требующие оценки ситуации человеком.



ТОМСКИЙ  
ПОЛИТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ

# Тестирование программного обеспечения

Копнов Максим Валериевич  
[kopnovmv@tpu.ru](mailto:kopnovmv@tpu.ru)

