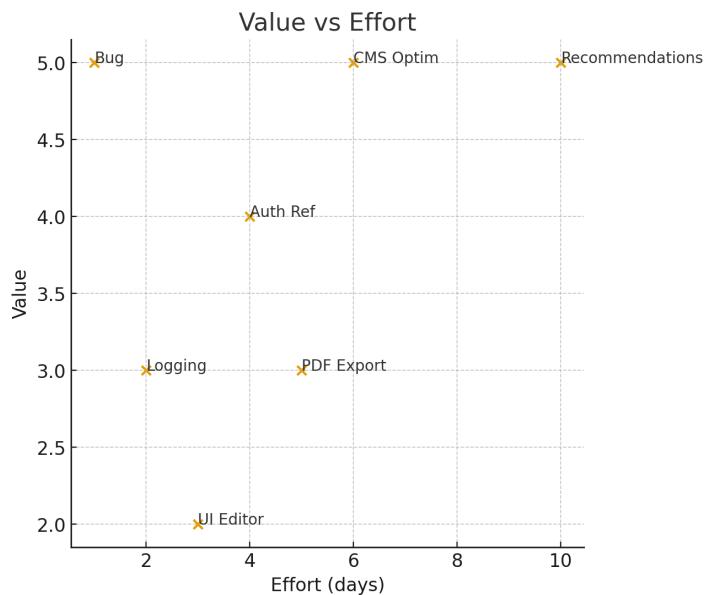


Практическая работа №5. Trade-off и метрики релиза

1. Матрица приоритетов



Анализ:

Высокая ценность, низкая сложность:

- Критический баг (Value 5 / Effort 1)
- Логирование (Value 3 / Effort 2)

Средняя ценность, средняя сложность:

- Авторизация (Value 4 / Effort 4)
- Оптимизация (Value 5 / Effort 6)

Высокая ценность, высокая сложность:

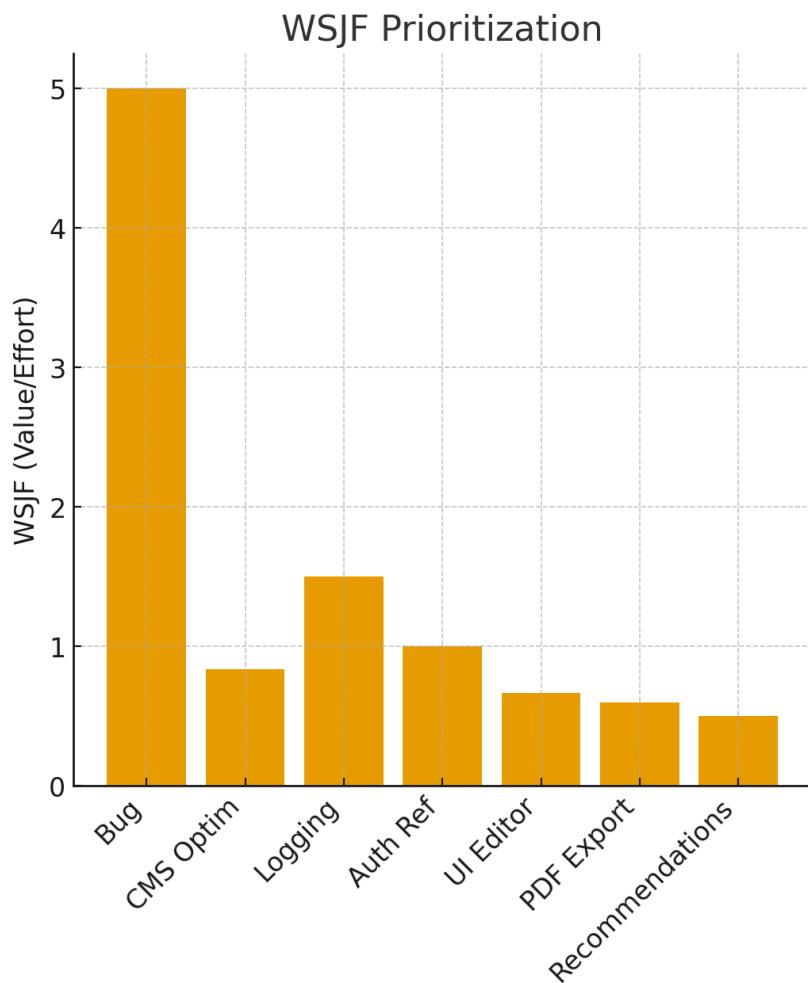
- Модуль рекомендаций (Value 5 / Effort 10)

Низкая ценность, средняя сложность:

- Интерфейс редактора
- PDF-экспорт

2. Расчет WSJF

WSJF = Value / Effort



3. Выбор задач, помещающихся в релиз

Доступно: 16 дней

Сначала выбираем максимальные WSJF:

1. Критический баг - 1
2. Логирование - 2

3. Авторизация - 4
4. Оптимизация - 6
5. Интерфейс редактора - 3

Итого: 16 дней

Задачи, которые переносятся:

- Модуль рекомендаций
- Экспорт в PDF

4. Расчет метрик релиза

Предположим:

- Стоимость дня разработчика: 15 000 р
- Стоимость дня тестировщика: 12 000 р
- Тестирование: 4 дня * 12 000 р = 48 000 р
- Инфраструктура релиза: 30 000 р

4.1 TCO (Total Cost of Ownership)

Разработка

$$16 \times 15\ 000 = 240\ 000 \text{ р}$$

Тестирование

$$48\ 000 \text{ р}$$

Инфраструктура

$$30\ 000 \text{ р}$$

$$\text{TCO} = 240\ 000 + 48\ 000 + 30\ 000 = 318\ 000 \text{ р}$$

4.2 CPU (Cost Per Unit)

Всего реализовано 5 задач:

$$\text{CPU} = \text{TCO} / 5$$

$$\text{CPU} = 318\ 000 / 5 = \mathbf{63\ 600\ р за задачу}$$

4.3 ROI

Прогнозируемая выгода релиза:

Задача	Ожидаемый эффект
Оптимизация скорости	+200 000 р
Критический баг	+150 000 р
Авторизация	+80 000 р
Логирование	+50 000 р
Интерфейс редактора	+40 000 р
Итого выгода	520 000 р

Формула ROI:

$$\text{ROI} = (\text{Выгода} - \text{TCO}) / \text{TCO}$$

$$\text{ROI} = (520\ 000 - 318\ 000) / 318\ 000$$

$$\text{ROI} = \mathbf{0.63\ (63\%)}$$

5. Черный лебедь

Сразу после старта релиза упали 3 производственных метрики:

- время ответа увеличилось на 25%

- ошибки 5xx увеличились на 17%
- нагрузка на CPU увеличилась на 35%

Причина:

Оптимизация скорости показала обратный эффект, изменение кэширующего слоя привело к деградации

Последствия:

- продукт требует срочного отката части релиза
- разработчики заменяют 6-дневную оптимизацию фиксом за 2 дня
- технический долг увеличился
- часть задач нужно исключить

Пересмотр приоритетов

Теперь доступно:

$$16 - 6 \text{ (оптимизация)} + 2 \text{ (фикс)} = \mathbf{12 \text{ дней}}$$

Переприоритизация:

Задача	Effort	Делать?
Критический баг	1	да
Логирование	2	да
Авторизация	4	да
Фикс для оптимизации	2	да

Интерфейс редактора	3	да
Оптимизация — отменена	6	нет
Рекомендации	10	нет
PDF	5	нет

Итог: 12 дней набираем ровно

Выводы:

Необходимо:

- делить крупные задачи на более мелкие, чтобы оптимизация не ломала всю систему
- ввести перед релизом обязательные нагрузочные, стресс- и регрессионные тесты по CPU/ARM
- ввести правило: глобальные core фичи только в релиз X+1
- ввести SLA на технический долг

Например:

- Не более 10% релиза -> техдолг
- Каждые 3 релиза -> техревизия
- увеличить наблюдаемость (observability). Логи + метрики + алерты -> чтобы деградация ловилась до релиза
- Ввести канареочный релиз
- Развернуть новую оптимизацию на 5% пользователей, увидеть деградацию заранее