

# Анализ SLA



Акимова Екатерина

Проверяемая работа: Курябов Данила

# В общем

- Работы демонстрируют понимание базовых понятий (релизный цикл, приоритизация, WSJF, TCO/ROI, наличие “чёрного лебедя”), но в текущем виде **не могут использоваться как управленческие артефакты: SLA/метрики неизмеримы, сроки реакции/фикса не привязаны к режиму поддержки, а PR4 и PR5 не согласованы по релизу/задачам.** Это создаёт риск принятия ошибочных решений о приоритетах и бюджете.

# Критерии оценки

Критерий оценки	Что оценивается
Контекст продукта и границы ответственности	Насколько понятно описан продукт, его компоненты (web, mobile, backend), внешние зависимости и распределение ответственности между командами
Корректность и реалистичность SLA и метрик	Корректное разделение SLA, SLO и KPI, реалистичность заявленных пороговых значений с учётом ресурсов
Измеримость метрик	Наличие окон измерения, источников данных и методов расчёта показателей
Операционная модель и работа с инцидентами	Определён ли режим поддержки (8×5 / 24×7), роли принятия решений, механизмы эскалации и отката релиза
Экономика и управленческие решения	Корректность расчётов TCO, CPU, ROI, прозрачность допущений и связь экономики с конкретным релизом

# Практическая работа №3 - «10 релизов / SLA как контракт»

**Что было хорошо:**

- Есть попытка вести “план/факт” по релизам и “чёрные лебеди” + действия (например, резервная зона после сбоя ДЦ).

+

•

○

# Недостатки

- Список SLA включает KPI: “удовлетворённость пользователей >4.2/5”, “количество багов в релизе <10”, “оценка пользователей” — это не внешняя гарантия сервиса, а продуктовые показатели/качество.
- Отклик <300мс — без уточнения: какой endpoint? p95? p99? на каком трафике? в какие часы?
- Реакция на баги <2ч — без модели поддержки. Если это 24×7, то нужен онколл минимум 5–7 инженеров в ротации; если 8×5 — тогда “<2ч” ночью/в выходные невыполнимо.

# Итог

Критерий	Балл (0–2)	Обоснование
Контекст продукта и границы ответственности	1	Продукт описан обобщённо, границы web/backend и внешних зависимостей не выделены явно
Корректность и реалистичность SLA	0	SLA смешаны с KPI (удовлетворённость, число багов), пороги не согласованы с ресурсами
Измеримость метрик	0	Не указаны окна измерения, источники данных и методы расчёта
Операционная модель	1	Есть реакции на инциденты, но нет режима поддержки и ролей принятия решений
Экономика и управленческие решения	2	Есть логика последствий инцидентов и действий команды
Итог	4 / 10	Учебный уровень, не готово к enterprise-применению

+

•

- **Практическая работа №4 – «Передача между продуктом и техникой: релизы X и X+1»**

Что сделано хорошо:

- Ясно перечислено, что именно продукт передаёт технике: требования, user stories, мокапы, бизнес-метрика, ограничения по срокам.
- Есть нефункциональные требования (latency, crash rate, критические баги).
- Есть реальные ограничения ресурсов (capacity): mobile 12 SP, web 8 SP.
- Есть “чёрный лебедь” с последствиями, включая конфликт web/mobile и срыв уточнения требований.




# Недостатки


- “SLA для тех. команды” фактически OLA/процессные договорённости, но это не обозначено: “оценка <2 дня”, “QA сборка за 5 дней”, “Support release notes за 24 часа”.
- Пересмотр SLA “в момент инцидента” выглядит как подгонка: “доля незапланированных задач 10% → 25%”- без правил пересмотра (error budget, период ревью, кто утверждает). Это P1, но для предприятия почти P0, потому что ломает доверие к SLA
- Нет DRI/Incident Commander/кто решает rollback.

# Итог


Критерий	Балл (0–2)	Обоснование
Контекст продукта и границы ответственности	<b>2</b>	Чётко описаны роли, команды и зоны ответственности
Корректность и реалистичность SLA	<b>1</b>	SLA фактически являются OLA, но это не обозначено явно
Измеримость метрик	<b>1</b>	Метрики перечислены, но не полностью операционализированы
Операционная модель	<b>0</b>	Нет DRI, Incident Commander, rollback-процедур
Экономика и управленческие решения	<b>1</b>	Учтены capacity и приоритеты, но без финансовой оценки
<b>Итог</b>	<b>5 / 10</b>	Хорошая структура, но слабый операционный контур



## Практическая работа №5 – «Trade-off и метрики релиза»



### Что хорошо:

- Есть матрица приоритетов и WSJF (Value/Effort) — это корректная механика выбора задач под ограничение.
  - Есть явный список, что помещается в релиз и что переносится, включая причины.
  - Есть расчёт TCO и CPU с разложением на разработку/тестирование/инфраструктуру.
  - Есть ROI с формулой и расчётом (пусть и с проблемами допущений).
- 

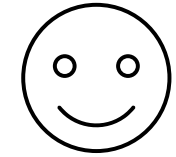
# Недостатки

- Экономика не согласована по допущениям и масштабу: “16 дней разработки × 15 000” — но не указан состав команды (это 1 разработчик 16 дней? или 4 разработчика 4 дня?). Для предприятия это принципиально.
- ROI считается из “ожидаемого эффекта +200k/+150k...” — но не указан горизонт (в месяц? в квартал?) и вероятность. Это P1 по чек-листу.
- Несостыковка с ПР4 по релизу (см. выше) — фактически trade-off сделан не для того релиза.

# Итог

Критерий	Балл (0–2)	Обоснование
Контекст продукта и границы ответственности	1	Контекст есть, но не синхронизирован с ПР4
Корректность и реалистичность метрик	2	Используются WSJF, TCO, CPU, ROI
Измеримость метрик	1	Формулы есть, но нет горизонта и источников
Операционная модель	1	Учтён чёрный лебедь, но без связи с поддержкой
Экономика и управленческие решения	0	Экономика не привязана к составу команды и релизу
<b>Итог</b>	<b>5 / 10</b>	Сильная идея, но слабая управленческая реализация

# Оценка: 5/10



Практическая	Оценка (1–10)	Что сильнее всего	Главный риск для предприятия
ПР3 (10 релизов / SLA)	4/10	Есть динамика релизов и реакции на «чёрных лебедей»	SLA смешан с KPI и не измерим
ПР4 (передача требований X и X+1)	5/10	Есть состав передаваемых данных, scarcity и инцидент	SLA фактически OLA, нет «кто решает» и режима поддержки
ПР5 (trade-off и метрики)	5/10	Есть WSJF + TCO/CPU/ROI + пересчёт после ЧЛ	Экономика и ROI без горизонта/команды, плюс несостыковка с релизом из ПР4