

СИСТЕМА АНАЛИЗА АВАРИЙНЫХ СИГНАЛОВ

СП «Татнефть-Добыча»

ПРОКСИМА

Проблематика

Масштаб

~60,000 аварийных сигналов в неделю

Основные причины ложных тревог

ПРИЧИНА	ВЛИЯНИЕ
Неверные уставки	40%
Изменение режимов работы	25%
Износ оборудования	20%
Выведенное оборудование	15%

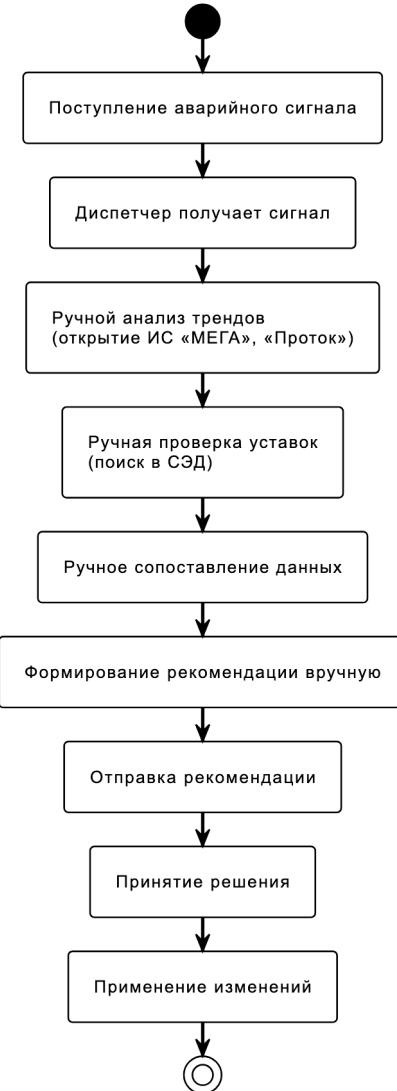
Текущий процесс

- Ручной анализ каждого сигнала
- Проверка трендов вручную
- Сопоставление с уставками вручную
- **Время анализа: несколько часов на объект**

Цель проекта

Автоматизировать анализ и снизить количество ложных тревог

Диаграмма: Текущий процесс



Фазовый подход к решению. Итерационная реализация

Поэтапное внедрение с накоплением данных

ФАЗА 1: Алгоритмическая система

- Детерминированные алгоритмы
- Анализ трендов и уставок
- Генерация рекомендаций
- Сбор статистических данных



ФАЗА 2: Система с ИИ

- Машинное обучение на собранных
- Прогнозирование отказов
- Самообучающиеся алгоритмы

ПРОКСИМА

Фаза 1: Результат

Интерфейсы для пользователей:

Роль	Интерфейс
Диспетчер ЕДС	Дашборд с текущими тревогами
Мастер ЦДНГ	Результаты анализа объектов
Руководство ЦДНГ	Управление уставками и рекомендациями
Специалист	Анализ рекомендаций по оборудованию

Что получаем:

- Веб-интерфейс для работы с системой
- API для интеграций
- Мобильное приложение (опционально)
- Отчеты и аналитика

Преимущества подхода

- Быстрый старт (без обучения моделей)
- Накопление данных для ИИ
- Снижение тревог уже на первой фазе
- Готовые интерфейсы для пользователей

Диаграмма: Фазовый подход. Алгоритмическая система

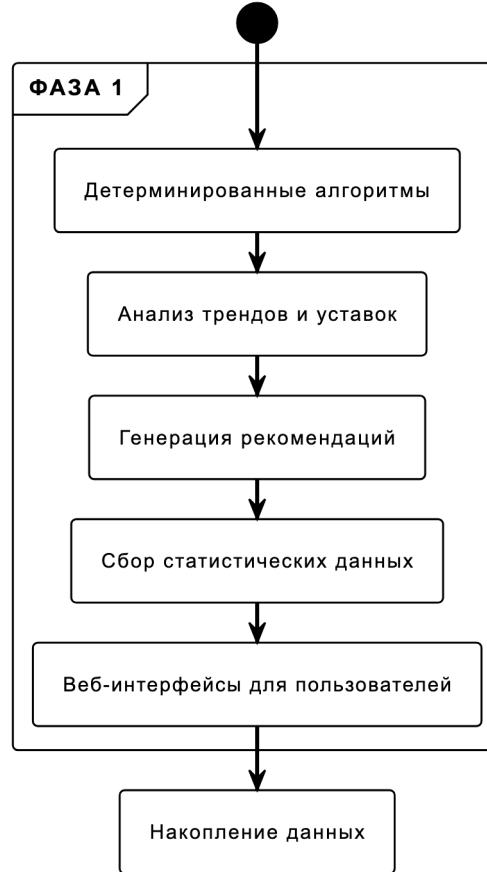
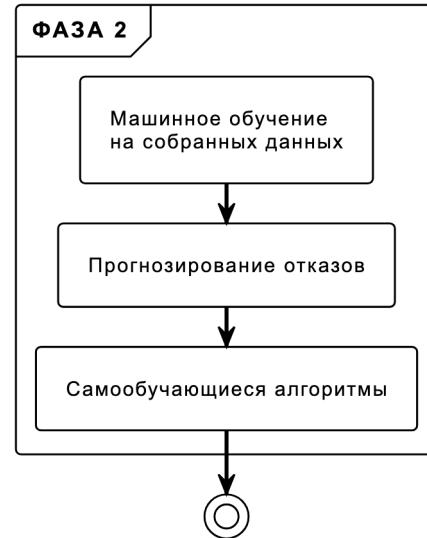


Диаграмма: Фазовый подход. Система с ИИ



Пример работающей системы - Поступление сигнала и анализ

Пример: "Высокое давление на приеме насоса Н-1"

Объект: ДНС-123 (НГДУ-1)

Этап 1: Поступление сигнала

ЕССД/АСДКУ → Система
"Аварийно-высокое давление: 0.42 МПа"

Роль: Диспетчер ЕДС

- Получает уведомление о сигнале
- Запускает анализ объекта через интерфейс

Этап 2: Автоматический анализ

Масштаб анализа:

- Анализ запускается для **группы объектов** (по НГДУ, типу объектов)
- Пример: все ДНС НГДУ-1 или топ-10 проблемных объектов
- **Один объект:** ~30 секунд
- **Группа объектов (10-20):** ~5-10 минут

Горизонтальное масштабирование:

- Параллельная обработка объектов
- Распределение нагрузки между серверами
- Масштабирование под нагрузку

Этап 2: Автоматический анализ

Система автоматически: 1. Собирает данные из внешних систем

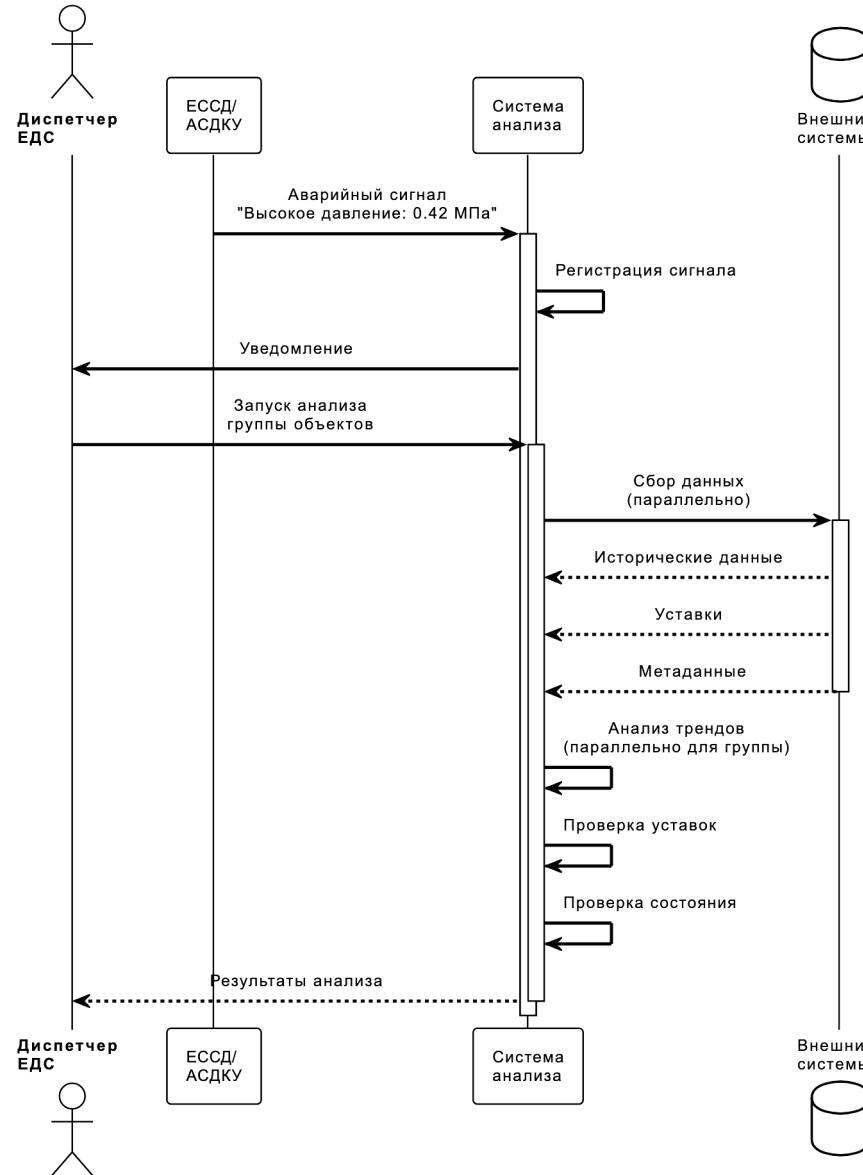
- Исторические данные (ИС «МЕГА», «Проток»)
- Объемы перекачки (КИС «Армитс»)
- Метаданные оборудования (1С ТОиР)
- Текущие уставки (СЭД)

Система автоматически: 2. Выполняет анализ (параллельно для группы)

- Анализ трендов (2 месяца)
- Проверка соответствия уставок
- Проверка состояния оборудования

Результат: Проблемы выявлены автоматически для всех объектов группы

Диаграмма: Процесс поступления сигнала и анализа



ПРОКСИМА

Алгоритм вычисления рекомендации

Шаг 1: Анализ трендов

Расчет статистики за 2 месяца:

ПАРАМЕТР	ТЕКУЩИЙ МЕСЯЦ	ПРЕДЫДУЩИЙ МЕСЯЦ	ИЗМЕНЕНИЕ
Среднее	0.34 МПа	0.33 МПа	+3%
Минимум	0.30 МПа	0.28 МПа	+7%
Максимум	0.42 МПа	0.40 МПа	+5%

Вывод: Давление стабильно, незначительный рост

Алгоритм вычисления рекомендации

Шаг 2: Проверка соответствия уставок

Сравнение фактических параметров с уставками:

Фактический максимум: 0.42 МПа

Текущая уставка: 0.40 МПа

Несоответствие: -0.02 МПа (уставка ниже!)

Проверка допустимых значений:

- Допустимый максимум (паспорт): 0.6 МПа
- Фактическое значение в пределах нормы

Алгоритм вычисления рекомендации

Шаг 3: Расчет рекомендуемой уставки

Формула расчета:

$$\begin{aligned}\text{Рекомендуемая уставка} &= \text{Фактический максимум} \times (1 + \text{Запас безопасности}) \\ &= 0.42 \text{ МПа} \times 1.10 \\ &= 0.46 \text{ МПа}\end{aligned}$$

Учет факторов:

- Запас безопасности: 10%
- Прогноз роста объемов: +20% → давление может вырасти
- Нормативные требования: соблюдены

Алгоритм вычисления рекомендации

Шаг 4: Определение приоритета и адресата

Приоритет: Высокий

- Уставка ниже фактического значения
- Регулярные ложные сработки (265/неделю)
- Влияние на производство

Адресат: Руководство ЦДНГ

- Требуется решение по изменению уставок
- Необходимо согласование с ООО «ПЦ»

Диаграмма: Алгоритм вычисления рекомендации. Шаг 1: Анализ трендов

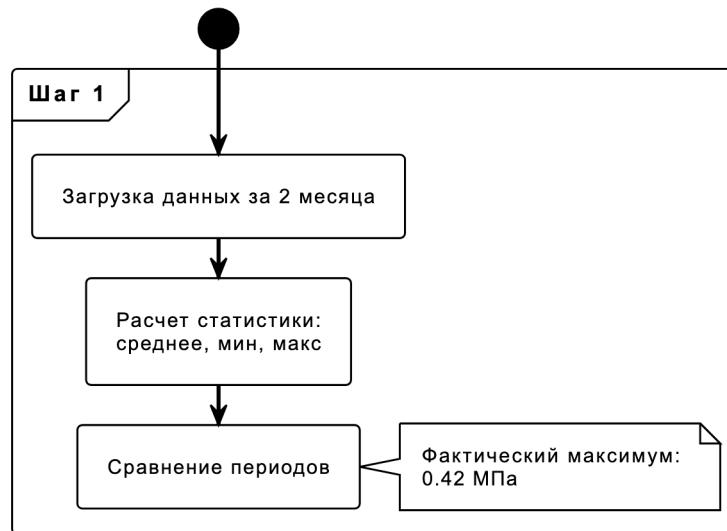


Диаграмма: Алгоритм вычисления рекомендации. Шаг 2: Проверка уставок

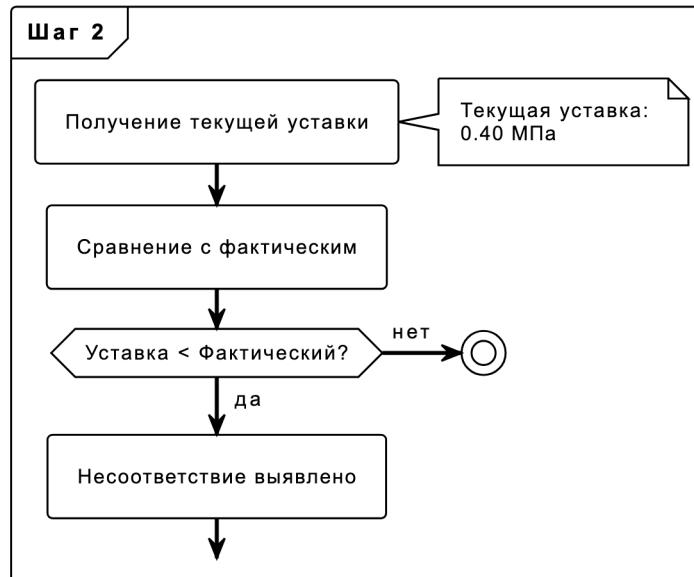


Диаграмма: Алгоритм вычисления рекомендации. Шаг 3: Расчет

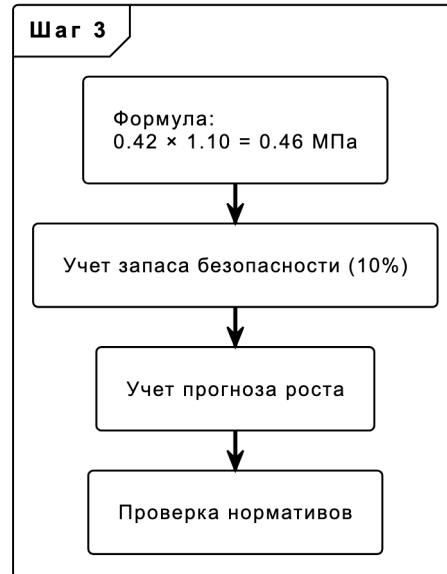
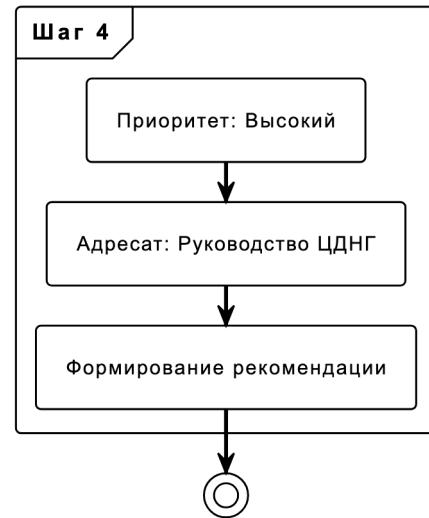


Диаграмма: Алгоритм вычисления рекомендации. Шаг 4: Определение



Результаты и рекомендации

Сгенерированная рекомендация

"Увеличить уставку с 0.40 до 0.46 МПа"

Краткое обоснование:

- Фактический максимум: 0.42 МПа
- Текущая уставка: 0.40 МПа (ниже фактического)
- Рекомендуемая: 0.46 МПа (с запасом 10%)
- Приоритет: **Высокий**

Результаты и рекомендации

Уведомления пользователям

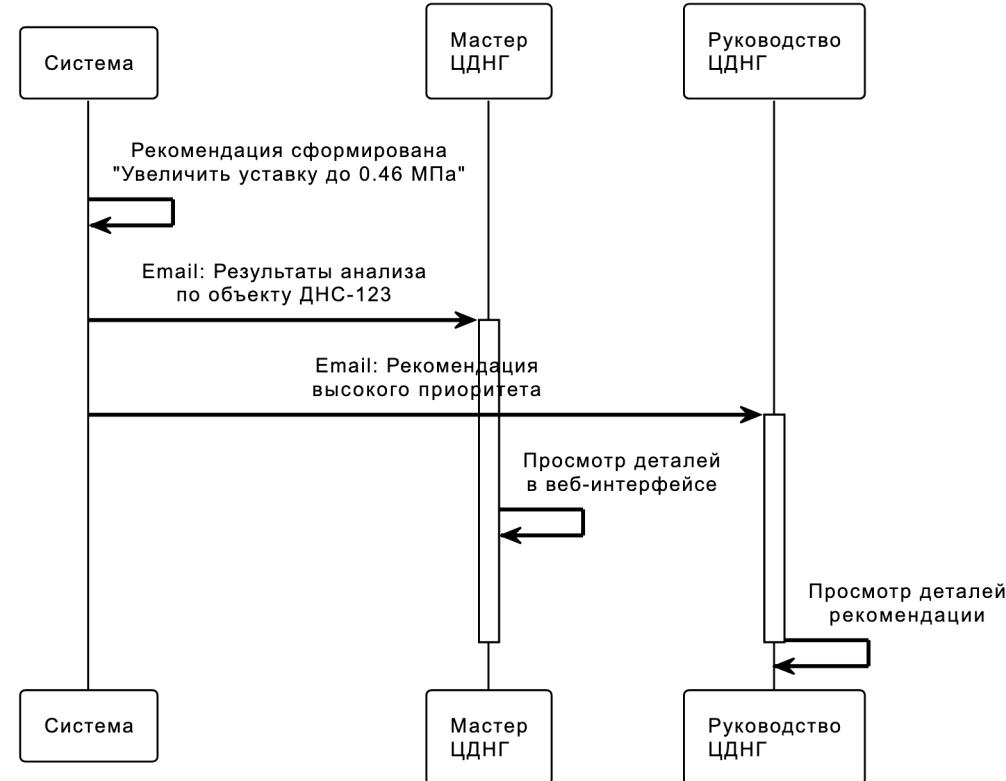
Роль: Мастер ЦДНГ

- Получает email с результатами анализа
- Просматривает детали в веб-интерфейсе

Роль: Руководство ЦДНГ

- Получает email о рекомендации высокого приоритета

Диаграмма: Уведомления пользователям



Принятие решения и применение рекомендации

Просмотр и принятие решения

Роль: Мастер ЦДНГ

- Изучает результаты анализа в интерфейсе
- Просматривает графики и статистику
- Информирует руководство

Роль: Руководство ЦДНГ

- Просматривает детали рекомендации
- Изучает обоснование
- **Принимает решение:** Принять рекомендацию

Принятие решения и применение рекомендации

Применение рекомендации

Роль: Руководство ЦДНГ

- Формирует заявку для ООО «ПЦ»
- Указывает новые значения уставок

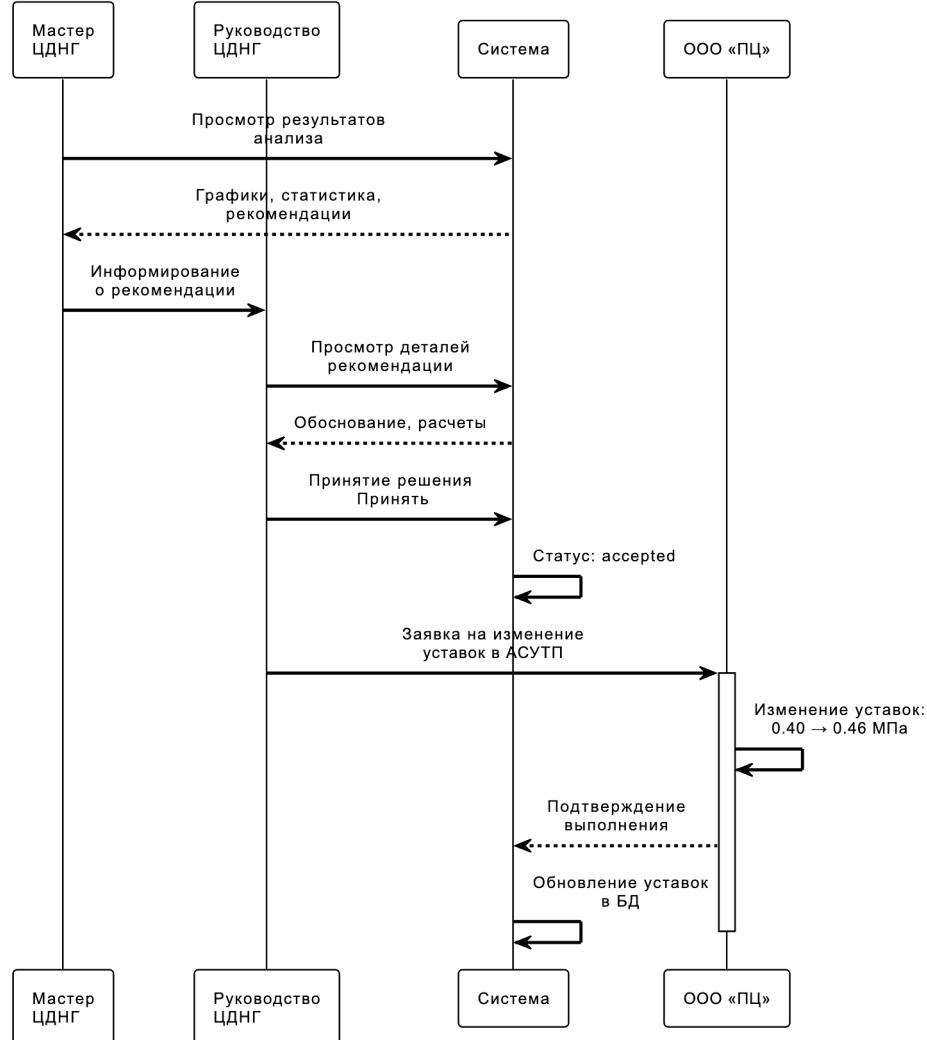
Роль: ООО «ПЦ»

- Изменяет уставки в АСУТП
- 0.40 МПа → **0.46 МПа**

Роль: Система

- Обновляет уставки в базе данных
- Отслеживает статус рекомендации

Диаграмма: Процесс принятия решения и применения



Результаты и эффективность

Мониторинг эффективности

Период проверки: 1-2 недели после применения

МЕТРИКА	ДО ПРИМЕНЕНИЯ	ПОСЛЕ ПРИМЕНЕНИЯ	РЕЗУЛЬТАТ
Сработок в неделю	265	15	↓ 94%
Ложные тревоги	250	0	↓ 100%

Результаты и эффективность. Итоги

Время обработки:

- Ручной анализ: **несколько часов** (на группу объектов)
- Автоматический анализ: **5-10 минут** (на группу объектов)
- **Ускорение:** в 10-20 раз

Результаты и эффективность. Итоги

Результаты:

- Проблема выявлена автоматически
- Рекомендация сформирована с обоснованием
- Решение принято на основе данных
- Эффективность подтверждена

Результаты и эффективность. Итоги

Роли, участвовавшие в процессе:

1. Диспетчер ЕДС - запустил анализ
2. Система - выполнила автоматический анализ
3. Мастер ЦДНГ - изучил результаты
4. Руководство ЦДНГ - принял решение
5. ООО «ПЦ» - применил изменения

Диаграмма: Результаты и эффективность

Мониторинг эффективности

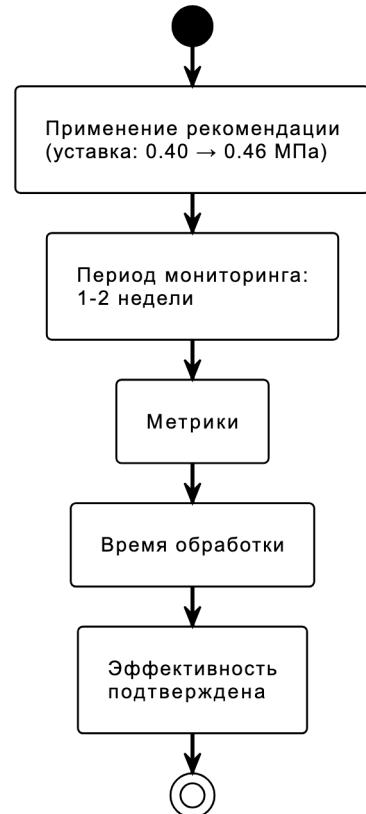
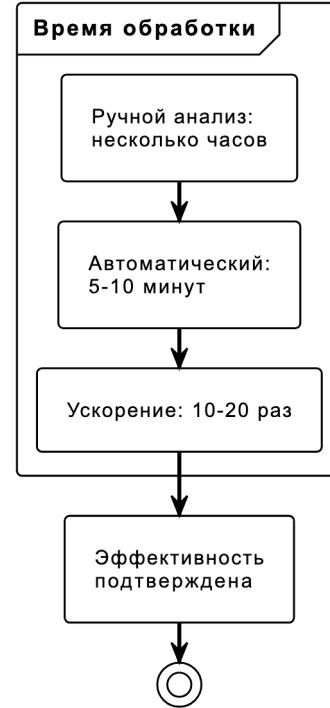


Диаграмма: Результаты и эффективность. Метрики



Диаграмма: Результаты и эффективность. Время Обработки



Отказоустойчивость и стабильность системы

Архитектурные решения для надежности

Горизонтальное масштабирование

- Множественные экземпляры сервисов
- Автоматическое масштабирование под нагрузку
- Распределение нагрузки между серверами
- **Результат:** Система выдерживает пиковые нагрузки

Отказоустойчивость и стабильность системы

Архитектурные решения для надежности

Резервирование компонентов

- Репликация баз данных (master-slave)
- Резервные копии данных
- Избыточность сервисов
- **Результат:** Отказ одного компонента не останавливает систему

Отказоустойчивость и стабильность системы

Архитектурные решения для надежности

Обработка ошибок и retry

- Автоматические повторы при сбоях внешних систем
- Circuit breaker для защиты от каскадных отказов
- Graceful degradation (деградация с сохранением функциональности)
- **Результат:** Система продолжает работать при сбоях интеграций

Отказоустойчивость и стабильность системы

Архитектурные решения для надежности

Асинхронная обработка

- Очереди задач для длительных операций
- Фоновые процессы для анализа
- Неблокирующая обработка запросов
- **Результат:** Высокая отзывчивость интерфейсов

Отказоустойчивость и стабильность системы

Архитектурные решения для надежности

Кэширование данных

- Кэш часто запрашиваемых данных
- Снижение нагрузки на внешние системы
- Быстрый отклик для пользователей
- **Результат:** Стабильная производительность

Отказоустойчивость и стабильность системы

Архитектурные решения для надежности

Мониторинг и алертинг

- Отслеживание состояния всех компонентов
- Автоматические уведомления о проблемах
- Метрики производительности
- **Результат:** Быстрое выявление и устранение проблем

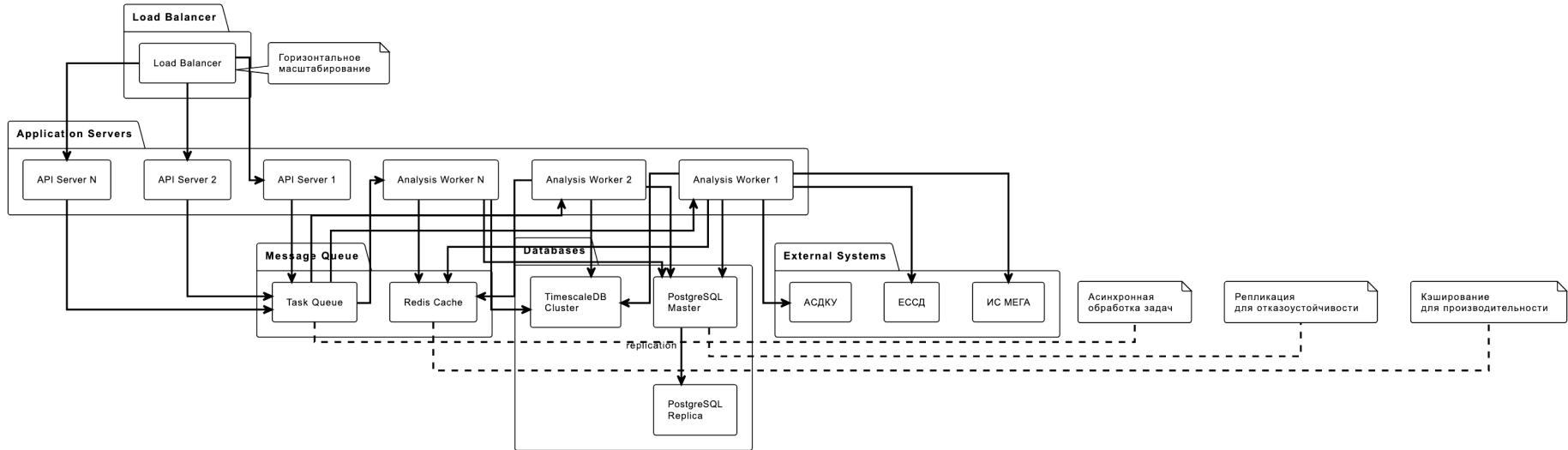
Отказоустойчивость и стабильность системы

Гарантии стабильности

- **Доступность:** 99.5% uptime
- **Производительность:** Обработка до 1000 объектов параллельно
- **Масштабируемость:** Автоматическое добавление ресурсов
- **Восстановление:** Автоматический перезапуск упавших сервисов

Отказоустойчивость и стабильность системы

Диаграмма: Архитектура отказоустойчивости



Фаза 1 - План реализации

Сроки реализации: 9-12 месяцев

ЭТАП	ДЛИТЕЛЬНОСТЬ	ОПИСАНИЕ
Анализ и проектирование	3-5 недель	Детализация требований, техническое проектирование
Разработка инфраструктуры	5-6 недель	БД, интеграции, базовое API
Разработка алгоритмов	6-8 недель	Trend Analyzer, Threshold Validator, Recommendation Generator
Разработка интерфейсов	6-8 недель	Веб-интерфейс, дашборды, страницы анализа
Интеграция и тестирование	5-6 недель	Интеграционное тестирование, приемочное тестирование
Пилотное внедрение	9-12 недель	Пилот на 5-10 объектах, доработка
Развертывание	6-9 недель	Масштабирование на все объекты ДНС, ГЗНУ

Фаза 1 - План реализации

Команда проекта

Роль	Количество	Обязанности
Аналитики	4	Анализ требований, проектирование процессов
Дизайнеры	2-4	Дизайн интерфейсов, UX/UI
Backend разработчики	6-8	Разработка API, алгоритмов, интеграций
Frontend разработчики	4-6	Разработка веб-интерфейса
QA/Тестировщики	4	Модульное, интеграционное, системное тестирование
DevOps инженеры	4	Инфраструктура, CI/CD, мониторинг, развертывание
Автоматические тестировщики	2-4	Настройка автоматических тестов, E2E тесты
Менеджер проекта	1	Управление проектом, координация

Фаза 1 - План реализации. Технологический стек

Backend:

- Python 3.11+, FastAPI, Celery
- PostgreSQL 15+, TimescaleDB, Redis
- Pandas, NumPy, SciPy

Frontend:

- React 18+, TypeScript 5+
- Material-UI, Chart.js/Recharts
- React Query, React Hook Form

Фаза 1 - План реализации. Технологический стек

Инфраструктура:

- Docker, Docker Compose
- Kubernetes (опционально)
- Nginx, Prometheus, Grafana
- GitLab CI/CD

Тестирование:

- pytest (Python)
- Jest, React Testing Library (Frontend)
- Playwright/Cypress (E2E)
- k6 (нагрузочное тестирование)

Фаза 1 - План реализации. Обоснование выбора технологического стека

Backend - Python:

- Лучшие библиотеки для аналитики (Pandas, NumPy, SciPy)
- Плавный переход к ML на Фазе 2 (scikit-learn, TensorFlow)
- Отличные инструменты для парсинга PDF документов
- Хорошая поддержка OPC UA интеграций

Базы данных:

- **TimescaleDB** - специализированная БД для временных рядов
- **PostgreSQL** - надежная реляционная БД с отличной поддержкой
- **Redis** - быстрый кэш и очереди задач

Фаза 1 - План реализации. Обоснование выбора технологического стека

Frontend - React + TypeScript:

- Современный стек с большой экосистемой
- TypeScript для надежности и масштабируемости
- Material-UI для быстрой разработки корпоративных интерфейсов
- Chart.js/Recharts для визуализации временных рядов

Инфраструктура:

- Docker/Kubernetes - стандарт для масштабируемости и отказоустойчивости
- Prometheus/Grafana - отраслевой стандарт мониторинга
- GitLab CI/CD - интеграция с процессом разработки

Диаграмма: Этапы разработки Фазы 1

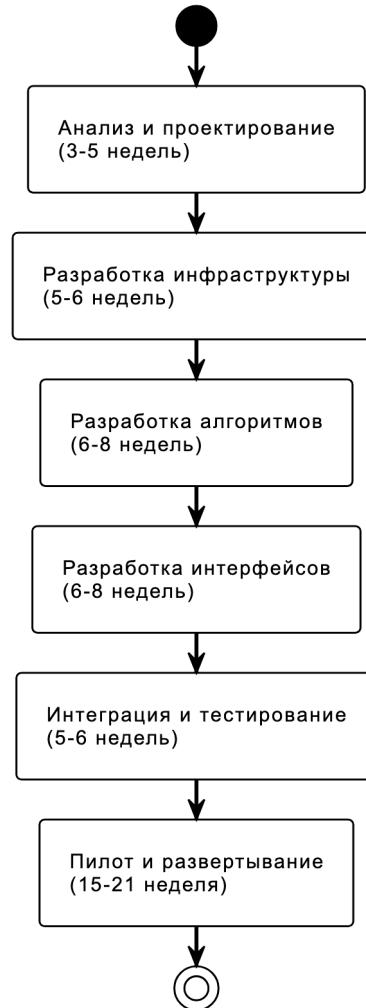


Диаграмма: Этапы разработки Фазы 1 (Анализ, проектирование и разработка инфраструктуры)

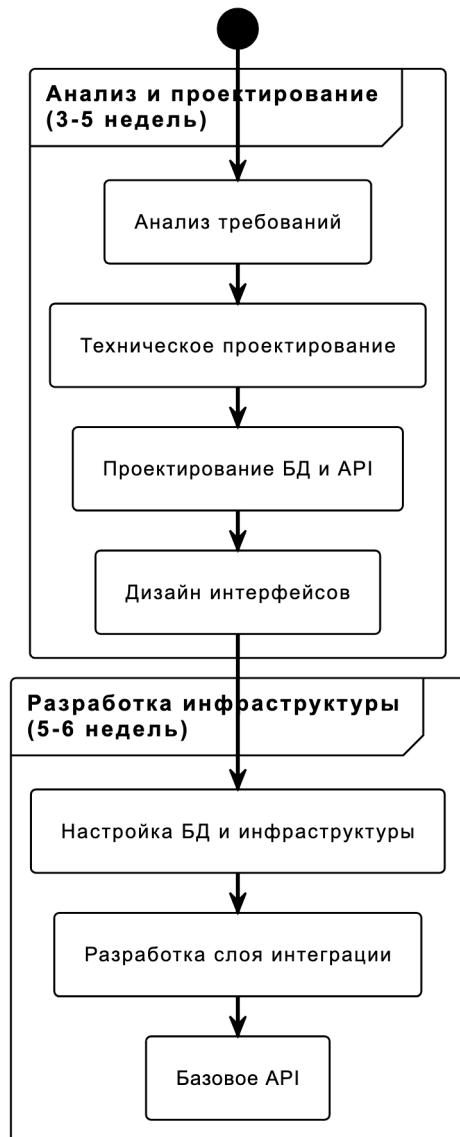
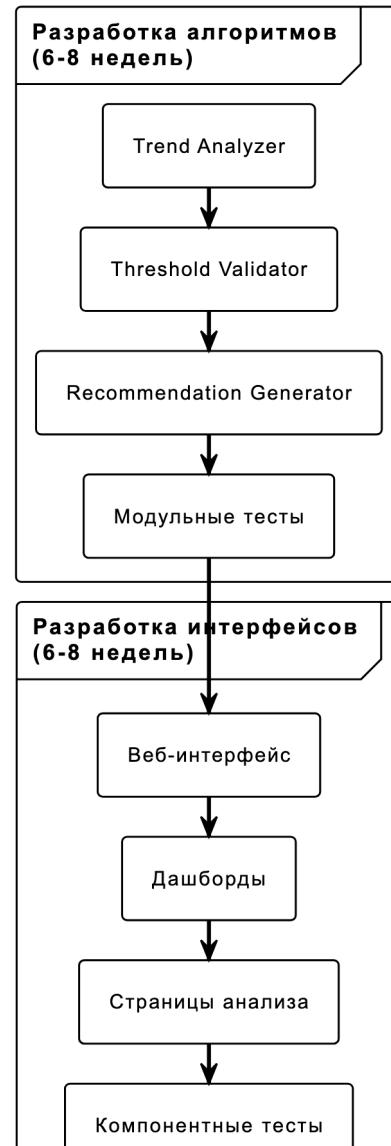


Диаграмма: Этапы разработки Фазы 1 (Разработка алгоритмов и разработка интерфейсов)



ПРОКСИМА

Диаграмма: Этапы разработки Фазы 1 (Интеграционное тестирование и развертывание)



ПРОКСИМА

Фаза 2 - План реализации

Сроки реализации: 6-9 месяцев

ЭТАП	ДЛИТЕЛЬНОСТЬ	ОПИСАНИЕ
Анализ данных и подготовка	5-6 недель	Анализ накопленных данных, подготовка датасетов
Проектирование ML-моделей	3-5 недель	Проектирование архитектуры моделей, выбор алгоритмов
Разработка ML-инфраструктуры	5-6 недель	ML Pipeline, обучение моделей, версионирование
Разработка моделей	9-12 недель	Обучение моделей, валидация, оптимизация
Интеграция ML в систему	5-6 недель	Интеграция моделей в существующую систему
Тестирование и валидация	5-6 недель	Тестирование моделей, А/В тестирование
Пилотное внедрение	6-9 недель	Пилот с ML-моделями, доработка
Развертывание	3-5 недель	Масштабирование ML-моделей

Фаза 2 - План реализации

Команда проекта

РОЛЬ	КОЛИЧЕСТВО	ОБЯЗАННОСТИ
Data Scientists	4-6	Разработка ML-моделей, анализ данных
ML Engineers	4	ML Pipeline, развертывание моделей
Backend разработчики	4-6	Интеграция ML в систему
QA/Тестировщики	2-4	Тестирование моделей, валидация
DevOps инженеры	2-4	ML инфраструктура, мониторинг моделей
Автоматические тестировщики	2	Автоматизация тестирования ML
Менеджер проекта	1	Управление проектом

Фаза 2 - План реализации. Технологический стек

ML/AI:

- Python 3.11+, scikit-learn
- TensorFlow/PyTorch (для глубокого обучения)
- MLflow (управление ML lifecycle)
- Pandas, NumPy для обработки данных

ML Infrastructure:

- Kubeflow или MLflow
- DVC (версионирование данных)
- Weights & Biases (эксперименты)
- Apache Airflow (оркестрация)

Фаза 2 - План реализации. Технологический стек

Backend (расширение):

- FastAPI (ML endpoints)
- Celery (асинхронная обработка)
- Redis (кэширование моделей)

Мониторинг ML:

- Prometheus (метрики моделей)
- Grafana (визуализация)
- Evidently AI (мониторинг дрифта данных)

Тестирование:

- pytest, pytest-ml
- Great Expectations (валидация данных)
- Model validation frameworks

Обоснование выбора технологического стека

ML/AI - Python экосистема:

- **scikit-learn** - стандарт для классического ML, отличная документация
- **TensorFlow/PyTorch** - лидеры в глубоком обучении, большой выбор моделей
- **Pandas, NumPy** - уже используются в Фазе 1, преемственность стека
- Единый язык для всего проекта (backend + ML)

ML Infrastructure:

- **MLflow** - управление ML lifecycle, эксперименты, версионирование моделей
- **Kubeflow** - оркестрация ML workflows в Kubernetes (если используется K8s)
- **DVC** - версионирование данных и моделей, интеграция с Git
- **Weights & Biases** - отслеживание экспериментов, визуализация

Обоснование выбора технологического стека

Интеграция с существующей системой:

- Использование тех же технологий (Python, FastAPI, PostgreSQL)
- Минимальные изменения в архитектуре
- Переиспользование инфраструктуры (Kubernetes, мониторинг)

Мониторинг ML:

- **Evidently AI** - специализированный инструмент для мониторинга дрифта данных
- **Prometheus/Grafana** - уже используются, единая система мониторинга

Диаграмма: Этапы разработки Фазы 2

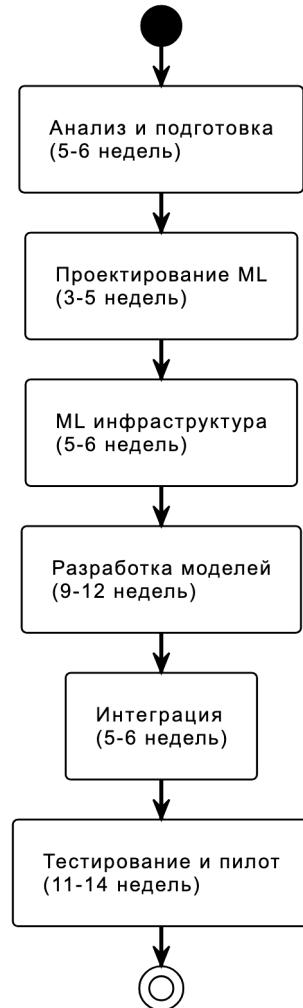


Диаграмма: Этапы разработки Фазы 2 (Анализ, подготовка, проектирование ML)

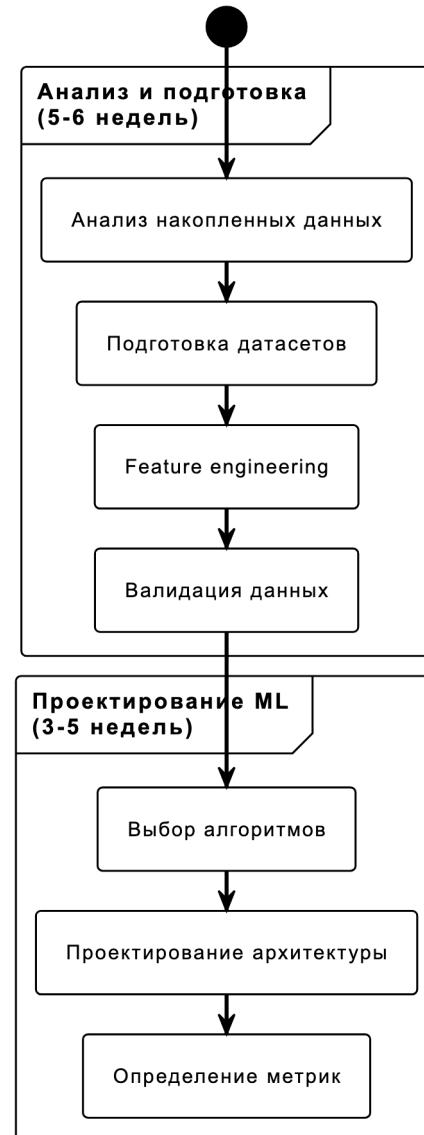


Диаграмма: Этапы разработки Фазы 2 (ML инфраструктура, разработка моделей)



Диаграмма: Этапы разработки Фазы 2 (Интеграция и тестирование)



ПРОКСИМА

Оценка стоимости

Общие сроки проекта

ФАЗА	СРОКИ	ОПИСАНИЕ
Фаза 1: Алгоритмическая система	9-12 месяцев	Разработка алгоритмической системы с интерфейсами
Фаза 2: Система с ИИ	6-9 месяцев	Разработка ML-моделей и интеграция
ИТОГО	15-21 месяц	Полный цикл разработки и внедрения

Оценка стоимости

Трудозатраты по фазам

ЭТАП	ДЛИТЕЛЬНОСТЬ	СРЕДНЯЯ КОМАНДА	ЧЕЛОВЕКО-МЕСЯЦЫ
Анализ и проектирование	3-5 недель	10 чел	2.0-2.6
Разработка инфраструктуры	5-6 недель	16 чел	5.0-6.0
Разработка алгоритмов	6-8 недель	12 чел	6.0-8.0
Разработка интерфейсов	6-8 недель	10 чел	6.0-8.0
Интеграция и тестирование	5-6 недель	16 чел	5.0-6.0
Пилот и развертывание	9-12 недель	12 чел	6.8-9.0
ИТОГО Фаза 1	9-12 месяцев	~12 чел (среднее)	30.8-39.6

ЭТАП	ДЛИТЕЛЬНОСТЬ	СРЕДНЯЯ КОМАНДА	ЧЕЛОВЕКО-МЕСЯЦЫ
Анализ данных и подготовка	5-6 недель	8 чел	2.6-3.0
Проектирование ML	3-5 недель	6 чел	1.2-1.8
ML инфраструктура	5-6 недель	10 чел	2.6-3.0
Разработка моделей	9-12 недель	12 чел	6.8-9.0
			ПРОКСИМА

Оценка стоимости

Итоговые трудозатраты

ПОКАЗАТЕЛЬ	МИНИМУМ	МАКСИМУМ	СРЕДНЕЕ
Человеко-месяцев	53.2	68.0	60.6
Человеко-часов	8,512	10,880	9,696
(при 160 часов/месяц)			

Оценка стоимости

Оценка стоимости проекта

КОМПОНЕНТ	ЧАСЫ	СТОИМОСТЬ (РУБ)
Фаза 1: Алгоритмическая система		
- Разработка	4,928 - 6,336	17,248,000 - 22,176,000
- Пилот и внедрение	1,088 - 1,440	3,808,000 - 5,040,000
Итого Фаза 1	6,016 - 7,776	21,056,000 - 27,216,000
Фаза 2: Система с ИИ		
- Разработка ML	3,584 - 4,544	12,544,000 - 15,904,000
- Пилот и внедрение	608 - 896	2,128,000 - 3,136,000
Итого Фаза 2	4,192 - 5,440	14,672,000 - 19,040,000
ИТОГО ПРОЕКТ	10,208 - 13,216	35,728,000 - 46,256,000
Средняя оценка	11,712	40,992,000

Оценка стоимости

Дополнительные расходы (не включены в основную оценку)

СТАТЬЯ	ОЦЕНКА (РУБ)
Инфраструктура (серверы, БД)	500,000 - 1,000,000/год
Обучение пользователей	300,000 - 500,000
Резерв на непредвиденные расходы (10%)	3,573,000 - 4,626,000

Итоговая оценка проекта

Разработка и внедрение:

- **Минимум:** 35,728,000 руб
- **Максимум:** 46,256,000 руб
- **Средняя:** 40,992,000 руб

С учетом дополнительных расходов:

- **Минимум:** ~40,000,000 руб
- **Максимум:** ~54,000,000 руб
- **Средняя:** ~47,000,000 руб

Ключевые метрики проекта

МЕТРИКА	ЗНАЧЕНИЕ
Общий срок	15-21 месяц
Средняя команда	10-12 человек
Трудозатраты	~61 человеко-месяцев
Стоимость разработки	~41.0 млн руб
Стоимость с доп. расходами	~47.0 млн руб

Заключение

Спасибо за внимание!