

# **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

---

Загрузка диаграммы...

# 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

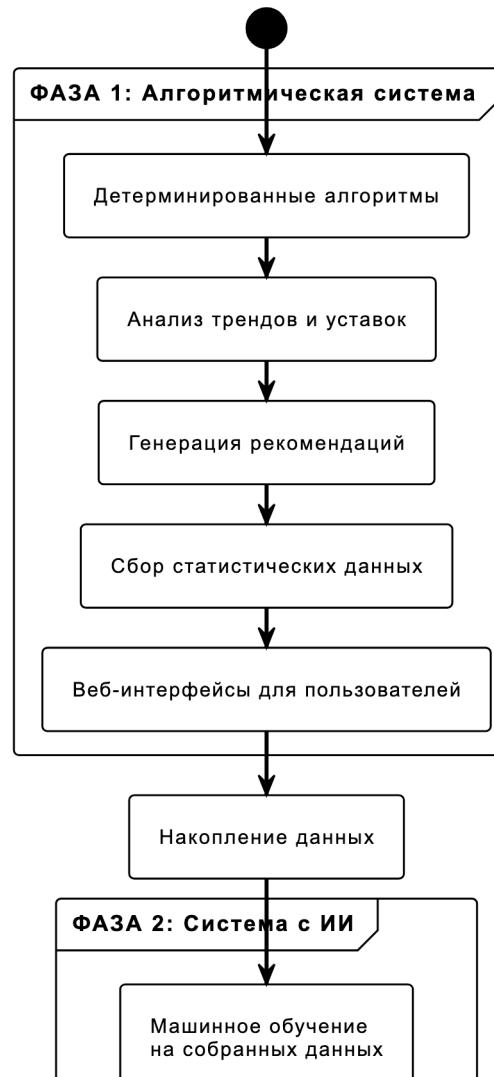
---

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

# Диаграмма: Фазовый подход

---

## Фазовый подход к реализации



ПРОКСИМА

# **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**

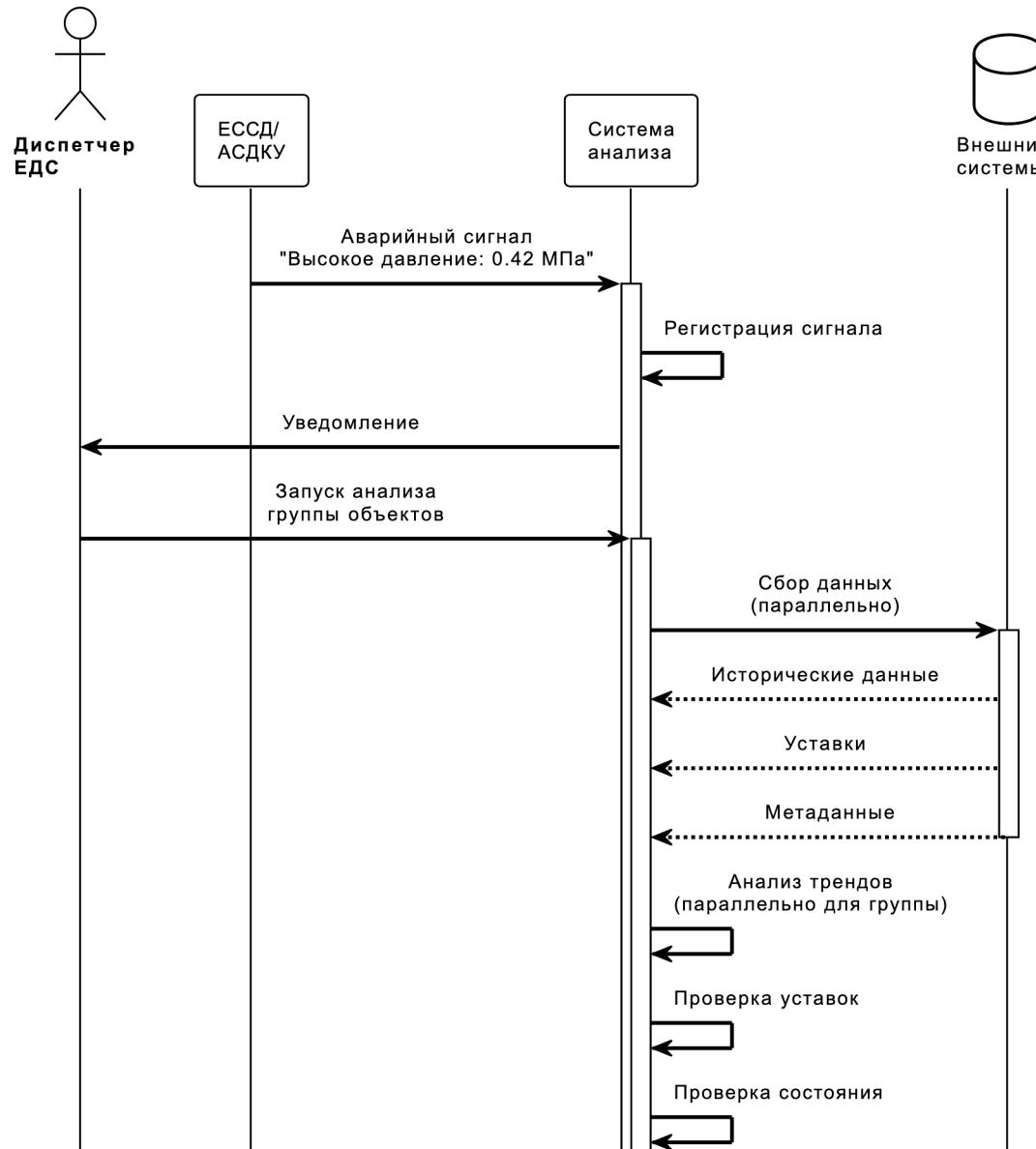
- Исторические данные (ИС «МЕГА», «Проток»)
- Объемы перекачки (КИС «Армитс»)
- Метаданные оборудования (1С ТОиР)
- Текущие уставки (СЭД)

**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

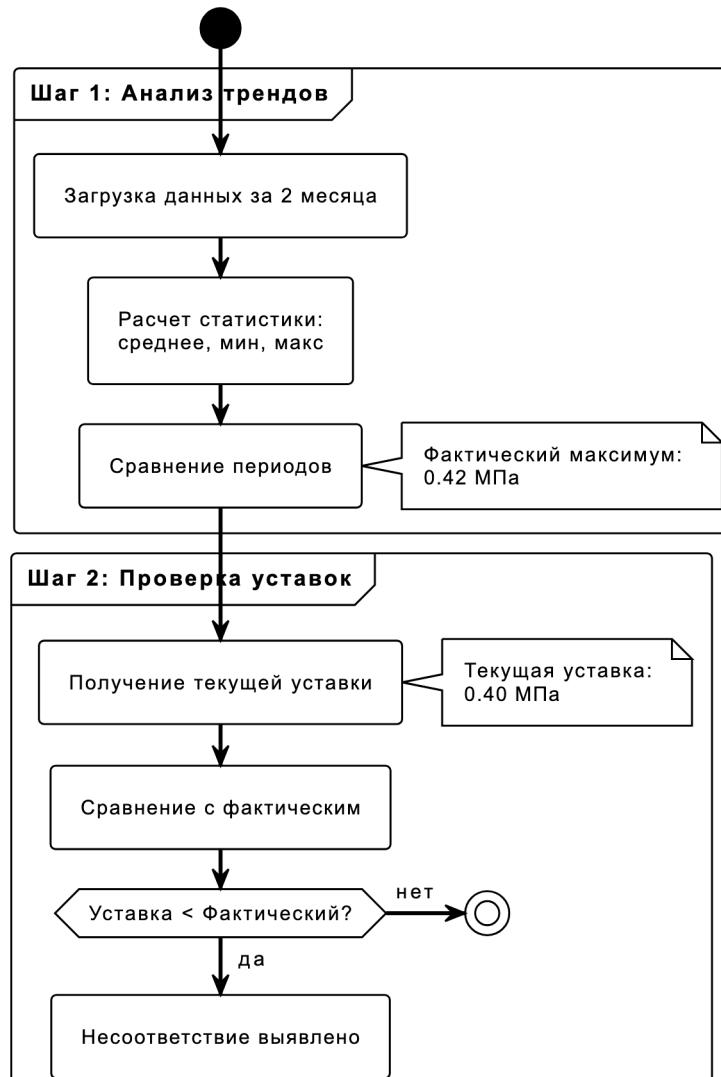
- Уставка ниже фактического значения
- Регулярные ложные сработки (265/неделю)
- Влияние на производство

Адресат: Руководство ЦДНГ

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

# Диаграмма: Алгоритм вычисления рекомендации

## Алгоритм вычисления рекомендации



ПРОКСИМА

# 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

---

Загрузка диаграммы...

# Принятие решения и применение

---

## 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

---

## 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

---

Загрузка диаграммы...

# 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

---

Загрузка диаграммы...

# 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

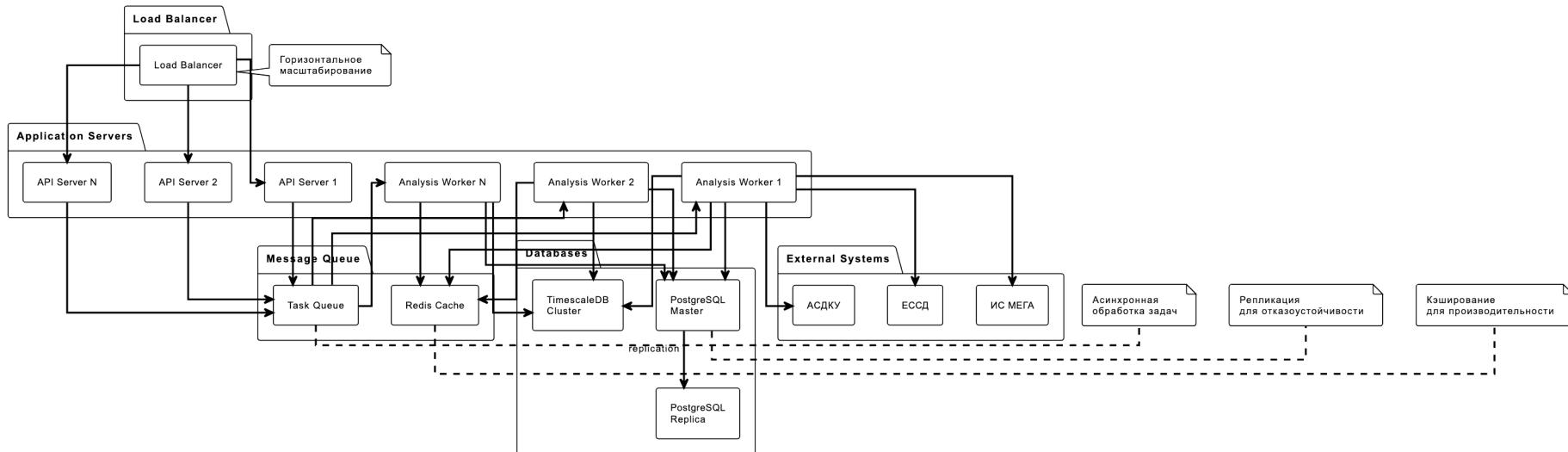
---

## 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
<b>Анализ и проектирование</b>	3-5 недель	Детализация требований, техническое проектирование
<b>Разработка инфраструктуры</b>	5-6 недель	БД, интеграции, базовое API
<b>Разработка алгоритмов</b>	6-8 недель	Trend Analyzer, Threshold Validator, Recommendation Generator
<b>Разработка интерфейсов</b>	6-8 недель	Веб-интерфейс, дашборды, страницы анализа
<b>Интеграция и тестирование</b>	5-6 недель	Интеграционное тестирование, приемочное тестирование
<b>Пилотное внедрение</b>	9-12 недель	Пилот на 5-10 объектах, доработка
<b>Развертывание</b>	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

### Инфраструктура:

- Docker, Docker Compose
- Kubernetes (опционально)

ПРОКСИМА

# 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

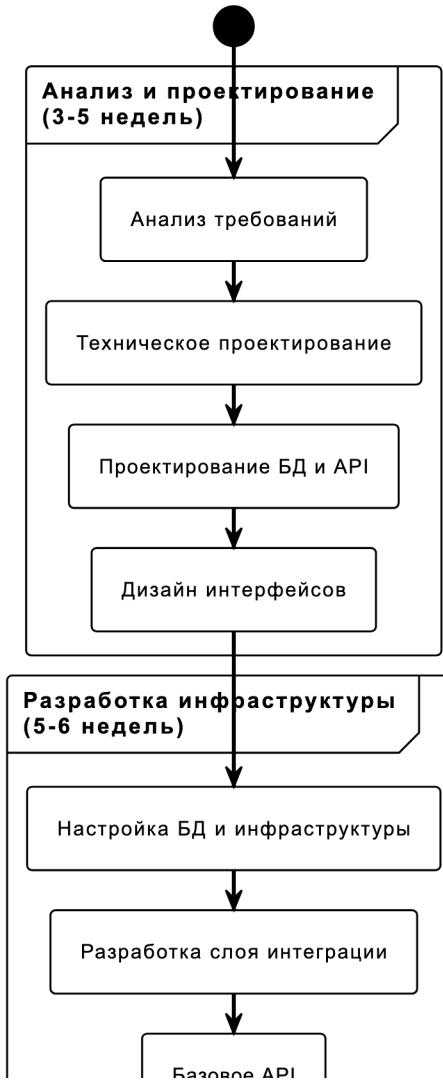
Frontend - React + TypeScript:

- Современный стек с большой экосистемой

# Диаграмма: Этапы разработки Фазы 1

---

## Фаза 1: Этапы разработки (9-12 месяцев)



ПРОКСИМА

## Фаза 2 - План реализации

---

**Сроки реализации: 6-9 месяцев**

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

### Backend (расширение):

ПРОКСИМА

# 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

## Интеграция с существующей системой:

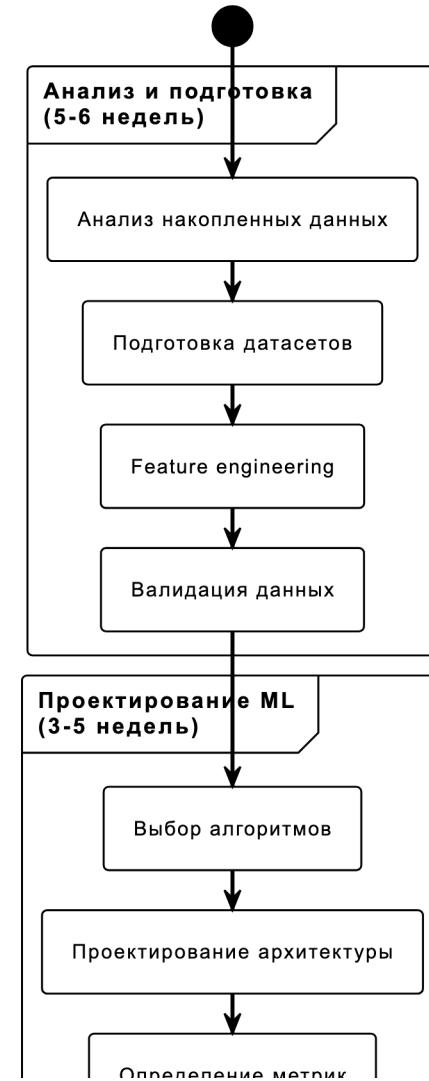
- Использование тех же технологий (Python, FastAPI, PostgreSQL)

ПРОКСИМА

# Диаграмма: Этапы разработки Фазы 2

---

## Фаза 2: Этапы разработки (6-9 месяцев)



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
<b>ИТОГО Фаза 1</b>	<b>9-12 месяцев</b>	<b>~12 чел (среднее)</b>	<b>30.8-39.6</b>

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

# 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!