ТЕХНИЧЕСКОЕ ЗАДАНИЕ

На разработку ИИ-системы интеллектуальной диагностики и прогнозирования состояния оборудования КИПиА

1. Общие сведения

Необходимо разработать программную систему с применением технологий искусственного интеллекта и машинного обучения для интеллектуального мониторинга, диагностики и прогнозирования технического состояния оборудования контрольно-измерительных приборов и автоматики (КИПиА).

Система предназначена для промышленных предприятий, эксплуатирующих большое количество датчиков, измерительных преобразователей, исполнительных механизмов, регулирующих органов и другого оборудования КИПиА.

В настоящее время значительная часть диагностики оборудования осуществляется на основании регламентных осмотров, периодических поверок, анализа аварийных сообщений и опыта инженерно-технического персонала. Такой подход позволяет выявлять часть неисправностей только после появления выраженных отклонений либо отказа оборудования.

Разрабатываемая система должна обеспечить переход от преимущественно реактивного и регламентного обслуживания к интеллектуальному мониторингу и прогнозному техническому обслуживанию (Predictive Maintenance).

Система должна получать технологические данные из промышленных информационных систем, анализировать временные ряды и взаимосвязи между параметрами, выявлять аномальные режимы работы, определять признаки деградации оборудования, оценивать вероятность потенциального отказа и формировать диагностические рекомендации для персонала.

Основная задача MVP — создание работающего программного модуля интеллектуальной диагностики для ограниченного набора типов оборудования КИПиА с возможностью последующего масштабирования на другие типы оборудования и технологические процессы.

Система не заменяет специалиста КИПиА и не выполняет автоматическое управление технологическим процессом. Решение является системой поддержки принятия решений.

Модуль поставляется как программный сервис с API и разворачивается в инфраструктуре заказчика либо в арендованном заказчиком облачном/серверном окружении на территории Республики Казахстан.

---

2. Термины и определения

Термин	Определение
КИПиА	Контрольно-измерительные приборы и автоматика
Оборудование	Датчик, измерительный преобразователь, исполнительный механизм или другое устройство КИПиА
Технологический параметр	Измеряемая величина: температура, давление, расход, уровень, вибрация и т.д.
Тег	Идентификатор технологического параметра в АСУ ТП/SCADA
Временной ряд	Последовательность значений параметра с временными отметками
Аномалия	Статистически или алгоритмически выявленное отклонение поведения параметра от нормального режима
Диагностическое событие	Выявленное системой отклонение, потенциальная неисправность или деградация оборудования
Прогноз отказа	Оценка вероятности возникновения неисправности в заданном временном горизонте
Остаточный ресурс	Оценка оставшегося периода эксплуатации оборудования до достижения заданного критического состояния
Уверенность	Числовая оценка достоверности результата модели от 0 до 1
Цифровой профиль	Совокупность технических, эксплуатационных и диагностических данных по конкретному оборудованию
Predictive Maintenance	Подход к техническому обслуживанию на основании прогнозируемого состояния оборудования

---

3. Границы работ (MVP)

Разрабатываемая система в рамках MVP должна обеспечивать:

- подключение к источникам технологических данных;
- импорт исторических данных;
- обработку временных рядов;
- контроль качества входных данных;
- выявление аномалий;
- определение отклонений от нормального поведения;
- диагностику типовых неисправностей;
- оценку вероятности потенциального отказа;
- формирование диагностических предупреждений;
- определение приоритета выявленного события;
- отображение текущего состояния оборудования;
- формирование цифрового профиля оборудования;
- сохранение истории диагностических событий;
- REST API;
- web-интерфейс для специалистов;
- журналирование работы системы;
- механизм обратной связи от специалиста КИПиА;
- формирование отчётности.

В MVP рекомендуется включить следующие основные классы оборудования:

C01 — датчики температуры

C02 — датчики давления

C03 — датчики расхода

C04 — датчики уровня

C05 — исполнительные механизмы и регулирующие клапаны

C06 — измерительные преобразователи

Архитектура системы должна предусматривать дальнейшее подключение других типов оборудования без полной переработки программного продукта.

Не входит в MVP

В рамках первой версии не предусматриваются:

- автоматическое управление технологическим процессом;
- самостоятельное изменение уставок ПИД-регуляторов;
- выдача управляющих команд исполнительным механизмам;
- замена существующей SCADA/АСУ ТП;
- физическое обслуживание и ремонт оборудования;
- автоматическая поверка приборов;
- управление промышленным оборудованием без подтверждения оператора;
- полная цифровая модель технологического предприятия;
- автономное принятие решений, влияющих на безопасность технологического процесса.

Данные функции могут быть включены в последующие версии продукта.

---

4. Входные данные

Система должна работать с историческими и текущими данными от промышленных информационных систем.

Возможные источники:

- SCADA;
- АСУ ТП;
- PLC/ПЛК;
- OPC UA;
- OPC DA;
- Modbus TCP;

- MQTT;
- REST API;
- промышленные базы данных;
- historian-системы;
- CSV/Excel-файлы;
- базы данных предприятия.

Основные входные данные:

- идентификатор оборудования;
- идентификатор технологического параметра;
- дата и время измерения;
- значение параметра;
- единица измерения;
- уставка;
- диапазон допустимых значений;
- аварийные и предупредительные пороги;
- состояние оборудования;
- события аварий;
- история ремонтов;
- история технического обслуживания;
- результаты поверок;
- информация о заменах оборудования.

Для обучения моделей могут использоваться исторические данные о фактических отказах оборудования и результатах технического обслуживания.

---

5. Типы интеллектуальной обработки

В систему включаются следующие основные функциональные классы.

Код	Модуль	Назначение	Приоритет
C01	Мониторинг	Контроль текущего состояния оборудования	A
C02	Контроль качества данных	Выявление пропусков, выбросов и некорректных данных	A
C03	Детектирование аномалий	Выявление нетипичного поведения параметров	A
C04	Интеллектуальная диагностика	Определение вероятного типа неисправности	A
C05	Прогнозирование отказа	Расчёт вероятности потенциального отказа	A
C06	Оценка состояния	Формирование индекса технического состояния	A
C07	Рекомендации	Формирование рекомендаций специалисту	A
C08	История оборудования	Хранение истории событий и обслуживания	A
C09	Предиктивное обслуживание	Формирование рекомендаций по срокам обслуживания	B
C10	Оценка остаточного	Расчёт Remaining Useful Life	B

	ресурса		
C11	Анализ взаимосвязей	Выявление взаимного влияния параметров	B
C12	Автоматическое переобучение	Адаптация моделей на новых данных	B

---

6. Функциональные требования

FR-1. Приём данных

Система должна принимать технологические данные в режиме реального времени и/или пакетном режиме.

Должна поддерживаться обработка исторических данных для обучения и тестирования моделей.

FR-2. Контроль качества данных

Перед передачей данных в AI-модели система должна выполнять:

- проверку временных меток;
- выявление пропусков;
- обнаружение дубликатов;
- обнаружение выбросов;
- проверку диапазонов;
- проверку единиц измерения;
- выявление зависших значений;
- выявление аномальной частоты обновления данных.

Некорректные данные не должны автоматически использоваться для формирования диагностического вывода без соответствующей маркировки.

FR-3. Формирование цифрового профиля оборудования

Для каждого прибора создаётся цифровой профиль:

- производитель;
- модель;
- серийный номер;
- тип оборудования;
- место установки;
- технологический объект;
- подключённые теги;
- диапазон измерений;
- дата установки;

- дата последней поверки;
- дата последнего обслуживания;
- история неисправностей;
- текущий технический статус;
- диагностические события.

FR-4. Мониторинг

Система должна отображать:

- текущее значение параметра;
- нормальный диапазон;
- предупредительный диапазон;
- критический диапазон;
- историю изменения параметра;
- текущий статус оборудования;
- диагностический индекс.

FR-5. Выявление аномалий

AI-модуль должен выявлять:

- резкие изменения параметров;
- постепенный дрейф;
- нестабильность показаний;
- повторяющиеся отклонения;
- несоответствие взаимосвязанных параметров;
- зависание значения;
- необычные циклы работы;
- отклонение от исторического поведения.

Каждое выявленное событие должно иметь:

- время;
- оборудование;
- параметр;
- тип аномалии;
- уровень критичности;
- значение;
- нормальный диапазон;
- оценку уверенности модели.

---

7. Интеллектуальная диагностика

AI-модуль должен определять вероятные причины выявленного отклонения.

Например:

Датчик давления

Система обнаруживает:

- постепенный дрейф показаний;
- нестабильность;
- скачки;
- зависание;
- несоответствие показаниям связанных параметров.

На основании исторических данных модель может сформировать результат:

Вероятность деградации датчика — 82%.

Вероятная причина — дрейф измерительного элемента.

Рекомендуемое действие — провести проверку и поверку прибора.

При недостаточном количестве данных система должна возвращать статус:

«Недостаточно данных для достоверной диагностики»

и не должна выдавать искусственно сформированный диагноз.

---

8. Прогнозирование отказов

Система должна оценивать вероятность отказа оборудования в заданном временном горизонте.

Например:

- вероятность отказа в течение 7 дней;
- вероятность отказа в течение 30 дней;
- вероятность отказа в течение 90 дней.

Результат может представляться следующим образом:

Параметр	Значение
Вероятность отказа	78%
Горизонт прогнозирования	30 дней
Текущее состояние	Предаварийное
Уверенность модели	91%
Приоритет	Высокий

---

9. Индекс технического состояния

Для каждого объекта должен рассчитываться интегральный индекс технического состояния от 0 до 100.

Пример:

90–100 — нормальное состояние

70–89 — удовлетворительное

40–69 — требуется наблюдение

20–39 — требуется техническая диагностика

0–19 — критическое состояние

Индекс должен рассчитываться на основании совокупности технологических параметров, истории оборудования и результатов AI-анализа.

---

10. Система рекомендаций

На основании результатов диагностики система должна формировать рекомендации для специалистов.

Примеры:

- проверить датчик давления;
- провести поверку;
- проверить импульсную линию;
- проверить кабельную линию;
- проверить питание;
- проверить исполнительный механизм;
- провести диагностику регулирующего клапана;
- увеличить частоту контроля;
- запланировать техническое обслуживание.

Рекомендации являются информационными и не должны автоматически приводить к изменению режима работы технологического оборудования.

---

11. Приоритизация

Каждому диагностическому событию присваивается приоритет:

P1 — критический

Высокая вероятность отказа и существенное влияние на технологический процесс.

P2 — высокий

Выраженные признаки деградации, требуется вмешательство специалиста.

P3 — средний

Отклонение требует наблюдения и дополнительной диагностики.

P4 — низкий

Незначительное отклонение, не требующее немедленного вмешательства.

---

12. Объяснимость AI-модели

Для каждого диагностического результата система должна по возможности показывать факторы, повлиявшие на вывод модели.

Например:

Причина повышения риска отказа:

• увеличение разброса показаний на 35%;

• устойчивый дрейф значения в течение 72 часов;

• увеличение количества выбросов;

• несоответствие показаний связанному технологическому параметру;

• аналогичное поведение наблюдалось перед предыдущим отказом.

Это необходимо для того, чтобы специалист КИПиА мог проверить вывод системы.

---

13. Уверенность модели

Каждый диагностический результат должен сопровождаться оценкой уверенности от 0 до 1.

Пример:

```
Диагноз:
Деградация датчика давления

Вероятность:
0.87
```

```
Уверенность:
0.91
```

Если уверенность ниже установленного порога, система должна отображать:

«Низкая уверенность — требуется проверка специалистом»

Система не должна выдавать предположительный диагноз как достоверный.

---

14. REST API

Система должна предоставлять REST API.

Предварительная структура:

```
POST /api/v1/data
```

Передача технологических данных.

```
POST /api/v1/equipment
```

Регистрация оборудования.

```
GET /api/v1/equipment/{id}
```

Получение цифрового профиля оборудования.

```
GET /api/v1/equipment/{id}/status
```

Получение текущего состояния.

```
GET /api/v1/equipment/{id}/diagnostics
```

Получение результатов диагностики.

```
GET /api/v1/equipment/{id}/forecast
```

Получение прогноза отказа.

```
GET /api/v1/events
```

Получение списка диагностических событий.

```
GET /api/v1/health
```

Проверка состояния сервиса.

API документируется в формате OpenAPI 3.

---

15. Пример результата API

```
{
  "equipmentId": "PT-101",
  "timestamp": "2026-11-26T12:00:00",
  "status": "warning",
  "healthScore": 62,
  "diagnostics": [
    {
      "type": "sensor_drift",
      "probability": 0.84,
      "confidence": 0.91,
      "priority": "P2",
      "message": "Выявлен устойчивый дрейф показаний датчика давления"
    }
  ],
  "forecast": {
    "failureProbability30d": 0.78
  },
  "recommendation": [
    "Провести проверку датчика",
    "Проверить импульсную линию",
    "Рассмотреть проведение внеплановой поверки"
  ]
}
```

---

16. Интерфейс пользователя

В MVP предусматривается web-интерфейс специалиста.

Основные разделы:

1. Главная панель мониторинга.
2. Перечень оборудования.
3. Карта состояния оборудования.
4. Карточка оборудования.
5. Графики технологических параметров.
6. Диагностические события.
7. Прогноз отказов.
8. Рекомендации.
9. История обслуживания.
10. Отчёты.

Главный экран должен позволять специалисту быстро определить:

- сколько оборудования работает нормально;
- сколько объектов требуют внимания;
- какие объекты находятся в критическом состоянии;
- какие отказы прогнозируются;
- какие мероприятия требуют выполнения.

---

17. Требования к качеству (KPI)

Основное требование — **не выдавать ложный уверенный диагноз**.

Система должна иметь возможность маркировать результат как неопределённый при недостаточном качестве данных.

KPI окончательно устанавливаются после формирования тестовой выборки.

Предлагаемые показатели MVP:

KPI	Целевое значение
Точность обнаружения аномалий	≥ 85%
Полнота обнаружения аномалий	≥ 85%
Precision диагностических событий	≥ 85%
F1-score диагностики	≥ 0,80
Точность прогноза отказа	≥ 80%
Ложная уверенная диагностика	≤ 5%
Доля корректно обработанных данных	≥ 99%
Доступность сервиса	≥ 99%
Время обработки нового события	≤ 5 сек
Время формирования расширенной диагностики	≤ 20 сек

Фактические значения KPI должны быть определены после анализа доступного набора данных и согласованы с заказчиком.

---

18. Технологическая архитектура

Предварительная архитектура решения:

Источник данных → интеграционный слой → хранилище → обработка временных рядов → AI/ML-модели → диагностический движок → API → Web-интерфейс.

Основные компоненты:

Data Layer

- PostgreSQL;
- TimescaleDB или аналогичное хранилище временных рядов;
- S3-совместимое хранилище.

Integration Layer

- OPC UA;
- Modbus TCP;

- MQTT;
- REST API;
- импорт CSV.

AI/ML Layer

- Python;
- PyTorch;
- Scikit-learn;
- XGBoost/LightGBM;
- модели anomaly detection;
- модели классификации;
- модели прогнозирования временных рядов.

Backend

- Python;
- FastAPI;
- REST API;
- Celery/RQ или аналогичный механизм фоновых задач.

Frontend

- React;
- TypeScript;
- графики временных рядов;
- dashboard.

Infrastructure

- Linux;
- Docker;
- Docker Compose;
- NVIDIA CUDA;
- GPU при необходимости обучения/инференса.

---

19. Безопасность

Система должна обеспечивать:

- авторизацию пользователей;
- разграничение прав доступа;
- TLS;
- API-ключи;
- ротацию ключей;
- журналирование действий;
- журналирование обращений к данным;
- резервное копирование;

- отсутствие передачи промышленных данных во внешние AI-сервисы без разрешения заказчика.

Для промышленных объектов предпочтительно использовать локальное или частное развёртывание.

---

20. Нефункциональные требования

Система должна обеспечивать:

- отказоустойчивость;
- масштабирование;
- возможность подключения новых источников данных;
- возможность добавления новых моделей;
- версионирование моделей;
- воспроизводимость результатов;
- журналирование;
- мониторинг;
- резервное копирование;
- восстановление после сбоя.

Архитектура должна позволять добавлять новые типы оборудования без изменения базового ядра системы.

---

21. Данные для обучения

Для разработки AI-моделей используются:

- исторические технологические данные;
- архивы SCADA;
- данные АСУ ТП;
- журналы аварий;
- журналы отказов;
- результаты поверок;
- результаты технического обслуживания;
- данные о ремонтах;
- экспертная разметка специалистов КИПиА.

Каждый набор данных должен иметь информацию о периоде, источнике и качестве данных.

Для обучения допускается использование открытых и синтетических наборов данных при условии соблюдения лицензий.

---

22. Обратная связь специалиста

Специалист должен иметь возможность отметить результат AI:

- «подтверждено»;
- «не подтверждено»;
- «ложное срабатывание»;
- «требуется дополнительная проверка»;
- «неисправность подтверждена».

Эти данные должны сохраняться и использоваться для последующего анализа качества модели и её переобучения.

---

23. Версионирование моделей

Каждая модель должна иметь:

- номер версии;
- дату создания;
- набор данных обучения;
- параметры обучения;
- показатели качества;
- дату внедрения;
- статус модели.

Например:

```
Model:
KIP-DIAG-1.0.0

Type:
Anomaly Detection

Equipment:
Pressure Sensors

Accuracy:
91.2%

F1:
0.87
```

Повторный анализ одних и тех же данных одной и той же версией модели должен обеспечивать воспроизводимый результат.

---

24. План развития

После завершения MVP система может быть расширена следующими функциями:

B01 — прогнозирование остаточного ресурса оборудования (RUL)

B02 — автоматическое определение причинно-следственных связей между параметрами

B03 — диагностика ПИД-контуров

B04 — диагностика регулирующих клапанов

B05 — интеллектуальная диагностика исполнительных механизмов

B06 — анализ вибрации оборудования

B07 — интеграция с ERP/EAM/ТОиР

B08 — автоматическое формирование заявок на ремонт

B09 — цифровой двойник оборудования

B10 — мультимодальная AI-модель для анализа технологических данных, документации и журналов эксплуатации

B11 — генеративный AI-помощник инженера КИПиА

B12 — прогнозирование экономических последствий отказа

---

25. Развёртывание

Система должна разворачиваться:

- на серверах заказчика;
- в частном облаке;
- в GPU-облаке на территории Республики Казахстан.

Внешние AI API не являются обязательными.

Все основные AI-модели должны иметь возможность работать в локальной инфраструктуре.

Поставка должна включать Docker-образы и инструкцию развёртывания.

---

26. Документация

Исполнитель должен предоставить:

- архитектурную документацию;

- API OpenAPI;
- инструкцию администратора;
- инструкцию пользователя;
- документацию по AI-моделям;
- описание структуры данных;
- описание алгоритмов;
- инструкцию развёртывания;
- документацию по резервному копированию;
- описание мониторинга;
- перечень программных зависимостей;
- информацию о лицензиях используемых компонентов.

---

27. Этапы разработки

Этап 1. Аналитика и подготовка данных

Срок: 2–3 недели.

Результат:

- анализ источников данных;
- описание архитектуры;
- формирование набора оборудования;
- подготовка датасета;
- определение KPI;
- разработка протокола разметки;
- прототип AI-модели.

Этап 2. Разработка MVP

Срок: 4–5 недель.

Результат:

- интеграционный модуль;
- хранилище;
- AI-модели;
- диагностический движок;
- API;
- web-интерфейс;
- система уведомлений;
- цифровой профиль оборудования.

Этап 3. Тестирование и внедрение

Срок: 2–3 недели.

Результат:

- тестирование;
- оптимизация моделей;
- проверка KPI;
- нагрузочное тестирование;
- развёртывание;
- документация;
- передача исходного кода и моделей.

Общий срок MVP: **около 10 недель**.

---

28. Результат проекта

Результатом проекта является программная AI-платформа для интеллектуального мониторинга и диагностики оборудования КИПиА.

В состав продукта входят:

1. AI-модели анализа технологических данных.
2. Модуль обнаружения аномалий.
3. Модуль диагностики неисправностей.
4. Модуль прогнозирования отказов.
5. Система оценки технического состояния.
6. Система рекомендаций.
7. Цифровой профиль оборудования.
8. API.
9. Web-интерфейс.
10. База данных.
11. Система журналирования.
12. Docker-образы.
13. Документация.
14. Исходный код.
15. Конфигурации и веса обученных моделей.

Главным технологическим результатом является переход от анализа оборудования по факту возникновения неисправности к **проактивному AI-мониторингу**, позволяющему выявлять признаки потенциального отказа до наступления аварийного события.