1. Введение

Данное техническое задание определяет требования к разработке подсистемы хранения данных для мониторинга износа режущих инструментов. Подсистема должна обеспечивать сбор, хранение, обработку и предоставление данных о состоянии инструментов, а также возможность управления данными (добавление, удаление, редактирование). Решение предназначено для интеграции с существующей системой мониторинга промышленного оборудования.

1. Цели и задачи

Цель: Создание надежной, масштабируемой и эффективной подсистемы для хранения данных о состоянии режущих инструментов.

Задачи:

* Обеспечение возможности хранения данных о различных типах режущих инструментов.
* Предоставление API для добавления, удаления, обновления и получения данных об инструментах.
* Обеспечение высокой доступности и отказоустойчивости.
* Обеспечение безопасности данных.
* Оптимизация хранения данных для повышения производительности запросов.
* Интеграция с существующей системой мониторинга (если применимо).
* Возможность масштабирования для поддержки растущего объема данных и количества инструментов.

1. Требования к функциональности
   1. **Сервис обработки данных:**

* Прием данных: Прием данных о состоянии инструментов (параметры износа, текущее состояние, идентификатор инструмента, время измерения и т.д.) по REST API или SignalR.
* Валидация данных: Проверка получаемых данных на соответствие заданным форматам и ограничениям.
* Сохранение данных: Сохранение данных в базу данных.
* Обработка запросов: Обработка запросов на получение, добавление, удаление и редактирование данных об инструментах.
* Управление инструментами
* Предоставление API: Предоставление REST API или SignalR интерфейса для взаимодействия с внешними системами.
* Логирование: Ведение подробного журнала событий (Serilog).
  1. **База данных:**
* Структура данных: Разработка структуры базы данных, обеспечивающей хранение информации об инструментах, их характеристиках, параметрах износа и истории изменений.
* Оптимизация запросов: Оптимизация запросов для быстрого получения данных.
* Резервное копирование: Реализация механизма резервного копирования данных.
* Восстановление данных: Реализация механизма восстановления данных.
* Безопасность данных: Обеспечение безопасности данных и доступа к ним (аутентификация, авторизация).
  1. **API:**
* Предоставление данных: Предоставление данных об инструментах в формате JSON или другом согласованном формате.
* Прием команд: Прием команд на добавление, удаление и редактирование данных.
* Аутентификация и авторизация: Обеспечение аутентификации и авторизации для доступа к API.
* Документация: Предоставление документации по API
  1. **Логирование**
* Запись событий: Запись всех важных событий в системе (ошибки, предупреждения, информационные сообщения).
* Конфигурация: Настройка уровней логирования и форматов записи.
* Хранение логов: Хранение логов в файл или централизованную систему логирования.
* Ротация логов: Реализация ротации логов для предотвращения переполнения диска.

1. Требования к реализации

Технологии (C#):

* Платформа: .NET 9.0.5 (C#)
* Сервис: ASP.NET Core Web API
* База данных: MS SQL Server
* ORM: Entity Framework Core
* API: REST API Логирование: Serilog

1. Требования к интеграции

* Подсистема должна легко интегрироваться с существующими компонентами системы мониторинга.
* Интеграция должна осуществляться через API.
* Необходимо предоставить примеры интеграции.

1. Тестирование

Необходимо провести всестороннее тестирование подсистемы, включая:

* Функциональное тестирование.
* Тестирование производительности.
* Тестирование безопасности.
* Тестирование надежности.
* Интеграционное тестирование.

1. Приемочные испытания

* Приемочные испытания должны проводиться на основе разработанных тестовых сценариев.
* Критерии приемки должны быть четко определены и задокументированы.

1. Документация

В процессе разработки необходимо подготовить следующую документацию:

* Техническое задание.
* Архитектурный проект.
* Схема базы данных.
* Описание API.
* Руководство пользователя.
* Руководство администратора.
* Результаты тестирования.

1. **Ответственность сторон**

* Исполнитель: Несет ответственность за разработку подсистемы в соответствии с требованиями данного технического задания и в согласованные сроки. Несет ответственность за качество кода и документации.
* Заказчик: Несет ответственность за предоставление необходимой информации и ресурсов для разработки подсистемы, а также за проведение приемочных испытаний и своевременную оплату работ.