

Техническое задание

На разработку подсистему хранения данных для системы обнаружения трещин в стеклянных изделиях

1. Введение

Подсистема хранения данных предназначена для обработки, хранения и управления информацией о трещинах, обнаруженных в стеклянных изделиях. Система должна обеспечивать надежное хранение данных, поддержку операций добавления, удаления и обновления информации, а также предоставлять интерфейс для взаимодействия с другими компонентами системы обнаружения трещин.

2. Цели и задачи

- Создать сервис для хранения и управления данными о трещинах.
- Обеспечить возможность добавления, удаления и редактирования данных.
- Обеспечить быстрый и надежный доступ к информации.
- Реализовать API для интеграции с системой обнаружения трещин и внешними приложениями.
- Обеспечить безопасность и целостность данных.
- Обеспечить масштабируемость и отказоустойчивость подсистемы.

3. Функциональные требования

- Прием и сохранение данных о трещинах (тип, размер, координаты, время обнаружения, степень опасности и др.).
- Поддержка CRUD-операций (создание, чтение, обновление, удаление).
- Поиск и фильтрация данных по различным параметрам.
- Ведение истории изменений данных.
- Логирование всех операций с данными.
- Резервное копирование и восстановление данных.
- Предоставление REST API для взаимодействия с другими системами.

4. Нефункциональные требования

- Надежность: устойчивость к сбоям, средняя наработка до отказа не менее 7500 часов.
- Безопасность: аутентификация и авторизация пользователей, защита данных.
- Производительность: минимальные задержки при обработке запросов.

- Масштабируемость: возможность расширения объема данных и числа пользователей.
- Совместимость с платформой .NET 9.0.5 и технологиями ASP.NET, Entity Framework.
- Использование PostgreSQL или MS SQL Server в качестве СУБД.
- Логирование с использованием Serilog.

5. Архитектура и технологии

- Язык программирования: C#.
- Платформа: .NET 9.0.5.
- Веб-сервис: ASP.NET.
- База данных: PostgreSQL.
- ORM: Entity Framework.
- Взаимодействие через REST API.
- Логирование: Serilog.

6. Структура данных

- Идентификатор трещины.
- Тип трещины (поверхностная, сквозная и т.п.).
- Геометрические параметры (длина, ширина, глубина).
- Координаты расположения на изделии.
- Дата и время обнаружения.
- Степень опасности.
- История изменений.

7. Требования к безопасности

- Контроль доступа на основе ролей.
- Шифрование данных при передаче.
- Защита от несанкционированного доступа.
- Регулярное резервное копирование.

8. План работ и сроки

- Анализ требований и проектирование – 1 неделя.
- Проектирование базы данных и API – 1 неделя.

- Разработка сервиса и интеграция с базой данных – 2 недели.
- Тестирование и отладка – 1 недели.
- Подготовка документации и внедрение – 1 неделя.

9. Документация

- Техническое задание.
- Руководство пользователя и администратора.
- ER-диаграмма базы данных.
- UML-диаграммы: варианты использования, последовательностей, компонентов, пакетов, деятельности.