бюджетное профессиональное образовательное учреждение Вологодской области

«Череповецкий лесомеханический техникум им. В.П. Чкалова»

Специальность **09.02.07** «Информационные системы и программирование»

**ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

**ПП по ПМ.05 Проектирование и разработка информационных систем**

Выполнил студент 3 курса группы ИС-32 Аксёнова Ксения Максимовна

подпись

место практики

наименование юридического лица, ФИО ИП Период прохождения:

с «25» мая 2025 г.

по «07» июня 2025 г.

Руководитель практики от предприятия должность

подпись МП

Руководитель практики от техникума:

Материкова А.А.

Оценка:

« » 2025 года

г. Череповец 2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Студент: Щепетова Оксана Евгеньевна

Группа: ИС-32

Место прохождения практики: ООО “Малленом Системс”

Руководитель практики: Материкова А.А.

Специальность: 09.02.07 — Информационные системы и программирование

Срок практики: 25.05.2025 по 07.06.2025.

Цель

Производственная практика направлена на формирование у обучающегося общих и профессиональных компетенцией, приобретение практического опыта и реализуется в рамках модуля ПМ.05 специальности

09.02.07 Информационные системы и программирование.

Задачи

1. Формирование общих компетенцией.
2. Формирование профессиональных компетенцией: проектирование и разработка информационных систем, выполнять анализ предметной области, основные модели построения информационных систем, платформы для создания, исполнения и управления информационной системой.
3. Приобретение практического опыта в: управлении процессом разработки приложений с использованием инструментальных средств,

обеспечении сбора данных для анализа использования и функционирования информационной системы, программировании в соответствии с требованиями технического задания, определения состава оборудования и программных средств разработки информационной системы, разработке документации по эксплуатации информационной системы.

ВЫПОЛНЯЕМЫЕ ЗАДАНИЯ

Задание 1

### **1) Изучить и проанализировать канбан-системы управления проектами**

**Кратко о методологии Канбан**:  
Канбан — это метод управления проектами, основанный на визуализации рабочих процессов. Он помогает контролировать поток задач и избегать перегрузки команды.

**Ключевые принципы Канбан:**

* Визуализация задач (доска, карточки, статусы)
* Ограничение WIP (Work In Progress — количество одновременно выполняемых задач)
* Управление потоком и постоянное совершенствование

**Популярные канбан-системы:**

| Система | Особенности |
| --- | --- |
| **Trello** | Простая, визуальная, бесплатная, с возможностью добавления плагинов |
| **Jira** | Инструмент для крупных проектов, интеграция с Agile и Kanban |
| **Asana** | Подходит для команд, визуально чистая, поддержка Канбан-доски |
| **ClickUp** | Комбинирует списки, доски, диаграммы Ганта, отлично для сложных проектов |
| **Notion** | Универсальный инструмент, в том числе с возможностью создания Канбан-досок |
| **YouTrack** | Альтернатива Jira, от JetBrains, гибкая система с Канбан-поддержкой |

### **2) Выбор подходящей системы управления проектами**

**Выбор: Trello**

**Почему Trello:**

* Простой интерфейс — подходит для студентов и небольших команд
* Бесплатный тариф с нужным функционалом
* Легко интегрируется с GitHub, Telegram и другими инструментами
* Хорош для MVP или учебного проекта

### **3) Реализация репозитория проекта**

Создаём репозиторий на GitHub:

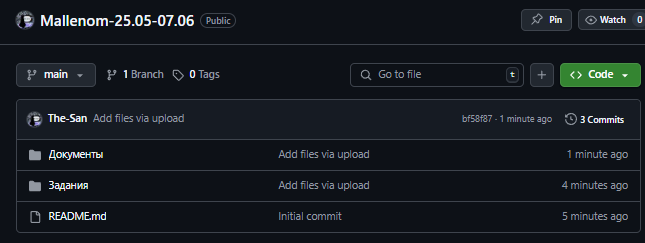


Рисунок 1 - Репозитории GitHab

Задание 2

**1) Выбор направления разработки**

**Выбор:** d. Подсистема хранения данных для мониторинга состояния подшипников

### Почему:

* Практическая значимость — мониторинг подшипников используется в промышленности и транспорте.
* Простая, но интересная структура данных (включает температуру, вибрации, износ, дату последнего обслуживания).
* Подходит для демонстрации CRUD-операций (добавление, удаление, обновление, просмотр состояния).

## **2. План работ + задачи в Канбан до окончания практики**

### Срок практики (пример): 2 недели (10 рабочих дней)

### План работ:

| Этап | Задачи | Срок |
| --- | --- | --- |
| 1. Анализ | Изучить предметную область (подшипники, параметры мониторинга) | 1 день |
| 2. Проектирование БД | ER-диаграмма, таблицы, связи | 1 день |
| 3. Разработка сервиса | Backend с CRUD (добавить/удалить/изменить/получить) | 3 дня |
| 4. Тестирование | Юнит-тесты, проверка API | 2 дня |
| 5. Документация | ТЗ, руководство пользователя и администратора | 1 день |
| 6. Подготовка к защите | Финальные правки, презентация, репетиция | 2 дня |

### Задачи на Канбан (Trello)

**Колонки:**

* To Do
* In Progress
* Done

**Примерные карточки задач:**

1. Изучить параметры мониторинга подшипников
2. Разработать ER-диаграмму
3. Создать модель БД в PostgreSQL
4. Настроить окружение (например, ASP.NET Core или Flask)
5. Реализовать добавление подшипников
6. Реализовать удаление подшипников
7. Реализовать обновление информации
8. Реализовать вывод списка подшипников
9. Написать тесты API
10. Составить Техническое Задание
11. Написать руководство пользователя
12. Написать руководство администратора
13. Проверка корректности данных и системы
14. Финальная сборка и деплой
15. Презентация проекта

## **3. Документация**

### a. **Техническое задание (ТЗ)**

**Название проекта:**  
Подсистема хранения данных для мониторинга состояния подшипников

**Назначение:**  
Обеспечить приём, хранение, удаление, изменение и просмотр данных о состоянии подшипников (температура, вибрация, дата последней проверки и т.д.)

**Требования:**

* Хранение данных в PostgreSQL
* CRUD-операции через REST API
* Валидация данных
* Документация API (Swagger)
* Авторизация
* Логирование запросов

**Функциональные требования:**

* Добавление нового подшипника
* Удаление подшипника по ID
* Обновление состояния подшипника
* Получение информации по одному/всем подшипникам

**Нефункциональные требования:**

* Надежность
* Устойчивость к некорректным данным
* Удобный интерфейс API

**Сроки реализации:** 3 недели

### b. **Руководство пользователя**

**Пользовательские возможности:**

* Получить список подшипников (GET /bearings)
* Просмотреть конкретный подшипник (GET /bearings/{id})
* Д обавить новый подшипник (POST /bearings)
* Изменить параметры подшипника (PUT /bearings/{id})
* Удалить подшипник (DELETE /bearings/{id})

### c. **Руководство администратора**

**Доступные действия:**

* Настройка базы данных
* Мониторинг логов (например, через journalctl или docker logs)
* Контроль целостности таблиц
* Создание бекапов и восстановление