

Отчёт по 2 этапу проекта

Сайт научного работника

Бойцов Александр Кириллович

Содержание

1	Цель работы	5
2	Выполнение работы	6
3	Выводы	10

Список иллюстраций

2.1	Файл об авторе	7
2.2	Файл для поста	8
2.3	Файл для публикации	9

Список таблиц

1 Цель работы

Добавить к сайту данные о себе.

2 Выполнение работы

Заполняю файл с данными о владельце сайта.

0 себе

👋 Меня зовут Александр Бойцов. Я первокурсник РУДН, обучаюсь по направлению "Бизнес-информатика" на физико-математическом факультете.

Мои интересы

- 📊 Цифровизация бизнес-процессов
- 💡 Информационные технологии в управлении
- 💻 Аналитика данных и автоматизация
- 🛡️ Внедрение IT-решений в компаниях

Чем занимаюсь

- 📖 Изучаю современные подходы в бизнес-информатике
- 📚 Применяю знания в учебных проектах
- 🌐 Исследую роль IT в развитии организаций

Открыт к новым знаниям и общению! 🚀

Рис. 2.1: Файл об авторе

Заполняю файл с текстом поста.

♦ Неделя открытий

Первая половина недели прошла под знаком ****практик по программированию****.

Писали небольшой проект на Python – симулятор очередей. Упор делали на функции, списки и ввод-вывод. Было полезно закрепить навыки работы с циклами и обработкой ошибок.

На парах по ****матанализу**** обсуждали предельные переходы и свойства непрерывности. Немного сложновато, но постепенно вырисовывается логика курса.

> Сделал для себя вывод: чем больше решаешь задач, тем меньше они кажутся пугающими.

Рис. 2.2: Файл для поста

Заполняю файл с текстом публикации.

Что такое CI/CD?

CI/CD – это практики современной разработки программного обеспечения, направленные на автоматизацию и ускорение жизненного цикла приложений. Акроним расшифровывается как:

- **CI** – *Continuous Integration* (*непрерывная интеграция*)
- **CD** – *Continuous Delivery* (*непрерывная доставка*) или *Continuous Deployment* (*непрерывное развертывание*)

Цель CI/CD – упростить и автоматизировать процесс разработки, тестирования и выката новых версий ПО без вмешательства вручную.

Непрерывная интеграция (CI)

Непрерывная интеграция означает, что каждый разработчик регулярно сливает (интегрирует) свои изменения в основную ветку репозитория. После каждого коммита запускаются автоматические процессы:

- Сборка проекта
- Прогон юнит-тестов
- Статический анализ кода
- Проверка на соответствие стилю

Таким образом, проблемы выявляются на раннем этапе, а не на стадии релиза.

--- --

Рис. 2.3: Файл для публикации

Перекомпилирую сайт

3 Выводы

Добавили к сайту данные о себе.