Министерство науки и высшего образования Российской Федерации

ФГАОУ ВО «Уральский федеральный университет   
имени первого Президента России Б.Н. Ельцина»

Институт новых материалов и технологий

Кафедра «Теплофизика и информатика в металлургии»

**Разработка контейнерного приложения Rust и настройка конвейера CI/CD с помощью GitHub Actions**

**ОТЧЕТ**

**по домашней работе**

**по дисциплине «Основы методологии Development Operation»**

Направление 09.03.02 «Информационные системы и технологии» (уровень бакалавриата)

Образовательная программа  
09.03.02/33.02 «Информационные системы и технологии» (СУОС)

Студент

группы НМТ-413901 Я.В.Крашенинников

Преподаватель:

профессор, д.т.н. В.В.Лавров

Екатеринбург

2025

Оглавление

[ЗАДАНИЕ 3](#_Toc198133336)

[РАЗРАБОТКА ПРИЛОЖЕНИЯ 3](#_Toc198133337)

[Разработка основного алгоритма 3](#_Toc198133338)

[Настройка конвейера 6](#_Toc198133339)

# ЗАДАНИЕ

Разработка контейнерного приложения Rust и настройка конвейера CI/CD с помощью GitHub Actions.

Приложение будет выполнять простые функции: сложение и вычитание.

# РАЗРАБОТКА ПРИЛОЖЕНИЯ

## Разработка основного алгоритма

Создание докер контейнера. Работаю в VS Code

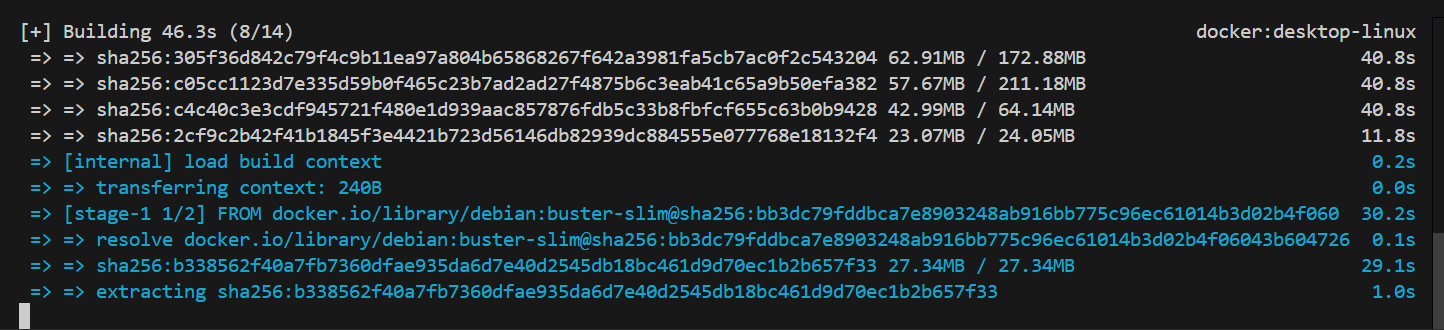


Рисунок 1 - docker build -t rust-ci-app .

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2 – main тестовый

Создаю зависимости для работы с JSON

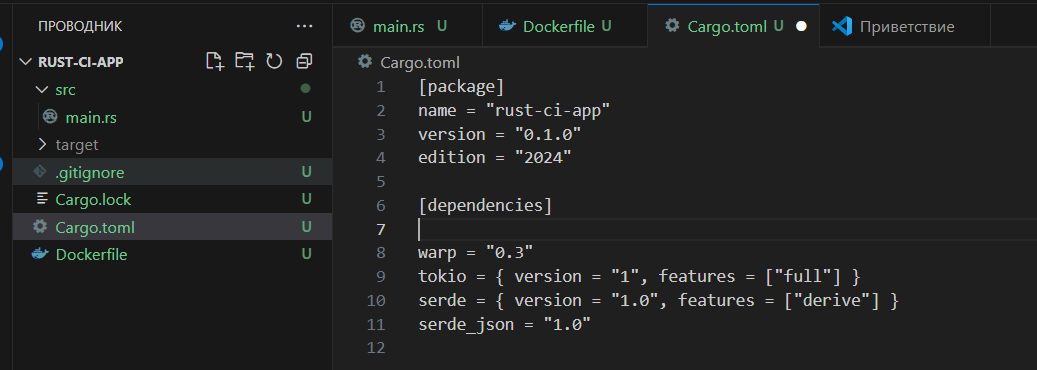


Рисунок 3 – Редактирую CARGO

Описываю логику расчетов в main

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 4 – описываю как и что считается

Сборка cargo

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 5 – процесс сборки

Проверка localhost

Изображение выглядит как текст, Шрифт, снимок экрана, Графика

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 6 – успешная работа на локале

Создание образа

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 7 – попытка сборки докера

Сборка контейнера

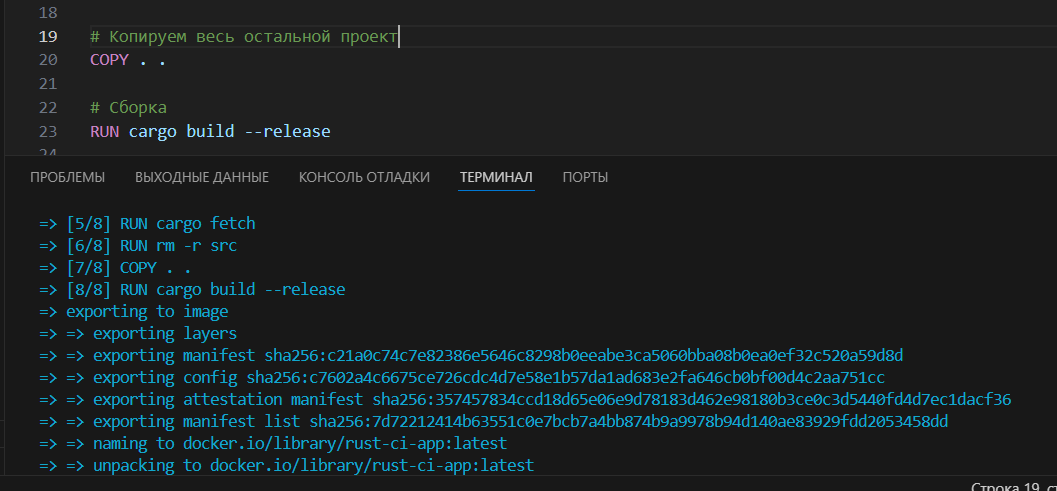


Рисунок 8 – успешная сборка

Запуск и проверка

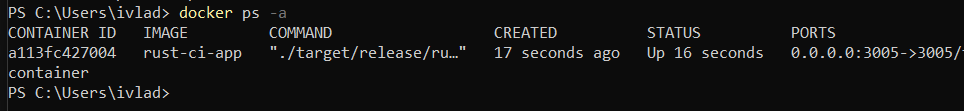


Рисунок 9 – проверка наличия контейнера

Проверим доступность с устройства на котором расположен контейнер Изображение выглядит как текст, Шрифт, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 10 – проверка по IP, работает

проверим с телефона

Изображение выглядит как электроника, текст, снимок экрана, Электронное устройство

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 11 – успешная работа на другом устройстве

## Настройка конвейера

Выгрузил все в гит через VS CODE, запушил

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 12 – публичная папка на гите

Настроим конвейер CI/CD с помощью GitHub Actions. Для этого выгрузим репозиторий и перейдем во вкладку “Actions” и выберем Docker image

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 13 – Автоматически созданный файл

Запустим Action

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 14 – Успешный старт и сборка

Таким образом был настроен автоматический сбор контейнера с помощью GitHub

Ссылка: https://github.com/krosshiik/homedeV.git