



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО
ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
В Г. ТАГАНРОГЕ РОСТОВСКОЙ ОБЛАСТИ
ПИ (филиал) ДГТУ в г. Таганроге

ОТЧЕТ

По дисциплине (модулю) Перспективные информационные технологии
(наименование учебной дисциплины (модуля))

на тему: "Инструменты моделирования IoT системы: распределённая система контроля версий Git."

Выполнил обучающийся: Дахно К.К.
(Ф.И.О.)

Направление:

09.03.02 Информационные системы и технологии
Код направления (наименование)

Обозначение отчет 2251853 Группа ВО ИСиТ-4122
номер зачетки

Проверил Доцент Орда-Жигулина М.В.
должность (Ф.И.О.)

Отчет защищён _____
дата _____
оценка _____
подпись _____

Таганрог
2025

Лабораторная работа 3.1

Инструменты моделирования IoT системы: распределённая система контроля версий Git.

Цель лабораторной работы: Познакомиться с инструментом Git, создать свой репозиторий на GitHub для хранения результатов лабораторных работ и освоить базовые операции: создание репозитория (init), фиксацию изменений (commit), ветвление (branch), слияние (merge) и синхронизацию с удаленным сервером (push/pull).

Постановка задачи:

1. Создать публичный репозиторий на платформе GitHub.
2. Загрузить структуру папок и отчеты по предыдущим работам через веб-интерфейс.
3. Освоить механизм Pull Request: внести изменения в код через новую ветку и выполнить слияние (Merge) с основной веткой.
4. Освоить работу через командную строку (консоль): клонировать репозиторий, изменить локальный файл симуляции и отправить изменения на сервер.

Используемые средства:

- Веб-сервис: GitHub.
- Клиент: Git Bash / VS Code Terminal.
- Система контроля версий: Git.

Ход работы:

1. Создание репозитория и загрузка материалов

Изм.	Лист	№ докум.	Подпись	Дат	ВО ИСиТ-4122			
Разраб.	Дахно К.К.				Перспективные информационные технологии	Лит.	Лист	Листов
Провер.	Орда-Жигулева							2
Н. Контр.	Орда-Жигулева							7
Утвёрд.	Орда-Жигулева					ПИ (филиал) ДГТУ в г. Таганроге		

Был создан новый публичный репозиторий с названием, соответствующим шаблону STUDY_FOG_group_.... В репозиторий добавлена структура директорий для лабораторных работ (Lab_3_1, Lab_3_2, Lab_3_3) и загружены соответствующие отчеты и скрипты.

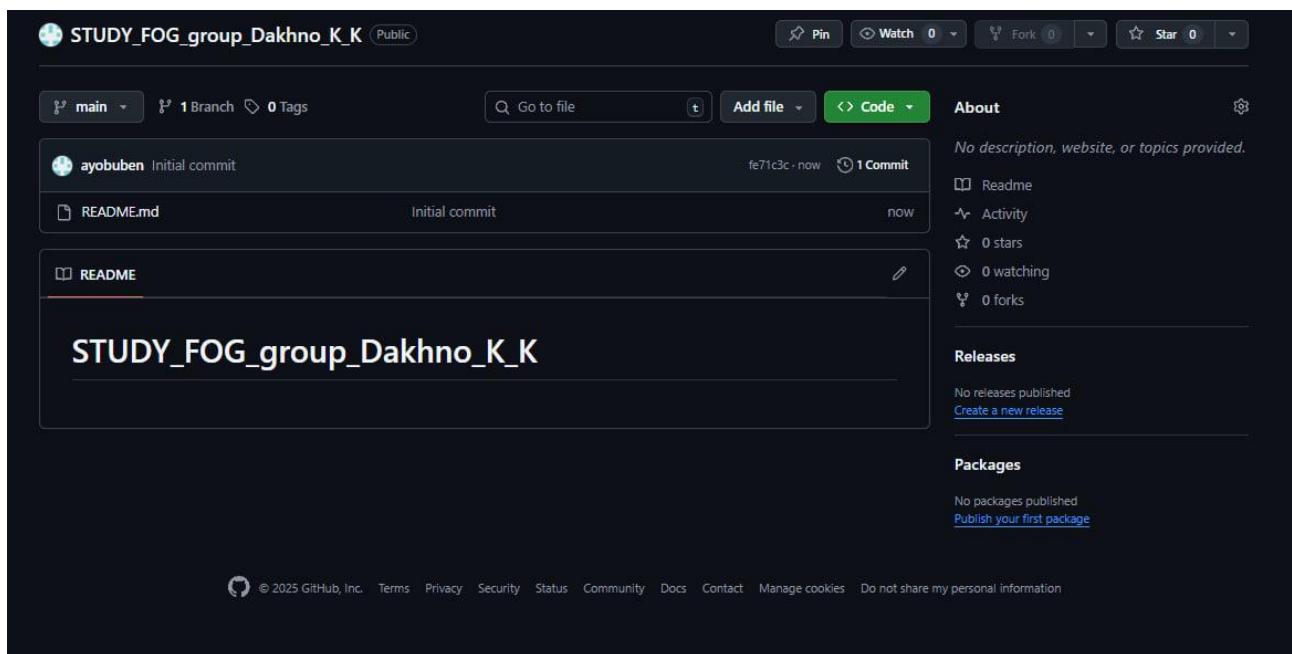


Рисунок 1 – Интерфейс созданного репозитория на GitHub



Рисунок 2 – Структура файлов проекта

Изм.	Лист	№ докум.	Подпись	Дата

УП.530000.000

Лист

7

2. Работа с ветками (Branching & Merging)

Для безопасного внесения изменений в файл lab2.py (оптимизация задержек согласно Лабораторной работе 3.2) был использован механизм ветвления.

1. Создана новая ветка patch-1.
2. Внесены правки в код (изменение диапазонов генерации случайных чисел random.randint).
3. Создан запрос на слияние (Pull Request).
4. После проверки изменений выполнено слияние (Merge) ветки patch-1 с основной веткой main.

```
Showing 1 changed file with 3 additions and 2 deletions.
diff --git a/labs/lab2.py b/labs/lab2.py
@@ -102,7 +102,7 @@ def simulate_buffer(n_tasks=30, read_interval_ms=120, title=""):
    print("== ЭКСПЕРИМЕНТ 1: ОПТИМИЗАЦИЯ ЗАДЕРЖЕК ==")
    # 1. Базовая модель (Fog 30-80)
-   avg_base = simulate_delay(fog_min=30, fog_max=80, title="Рис. 1 - Базовая сквозная задержка")
+   avg_base = simulate_delay(fog_min=10, fog_max=40, title="Рис. 1 - Базовая сквозная задержка")
    print(f"Базовая средняя задержка: {avg_base:.2f} мс")
    # 2. Оптимизированная модель (Fog 10-40)
@@ -118,4 +118,5 @@ def simulate_buffer(n_tasks=30, read_interval_ms=120, title=""):
    simulate_buffer(read_interval_ms=60, title="Рис. 3 - Быстрое чтение (буфер пуст)")
    # 4. Медленное чтение (200 мс)
-   simulate_buffer(read_interval_ms=200, title="Рис. 4 - Медленное чтение (накопление)")
+   simulate_buffer(read_interval_ms=200, title="Рис. 4 - Медленное чтение (накопление)")
```

Рисунок 3 – Просмотр изменений в Pull Request

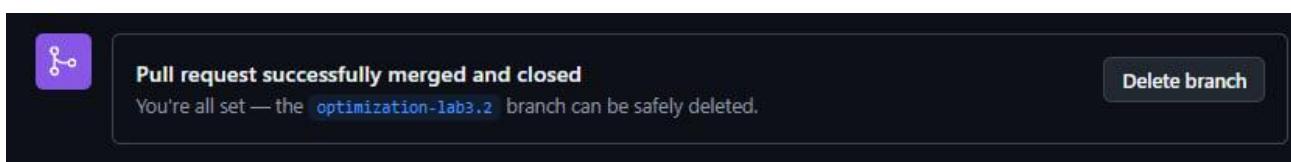


Рисунок 4 – Успешное слияние веток

3. Работа через командную строку

Для автоматизации работы с файлом lab3.py использовались консольные команды Git.

1. Выполнено клонирование удаленного репозитория на локальную машину командой: git clone https://github.com/...

Изм.	Лист	№ докум.	Подпись	Дата

УП.530000.000

Лист

7

2. В локальной среде разработки (VS Code) файл симуляции был обновлен кодом для Варианта №1 (Лабораторная работа 3.3).
3. Изменения зафиксированы и отправлены на сервер командами:
 - git add . (добавление в индекс)
 - git commit -m "Update lab 3.3" (создание снимка версии)
 - git push (отправка на GitHub)

```
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 16 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 446 bytes | 446.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/ayobuben/STUDY_FOG_group_Dakhno_K_K.git
  4d56c7a..1d6008b  main -> main
```

Рисунок 5 – Работа с репозиторием через терминал

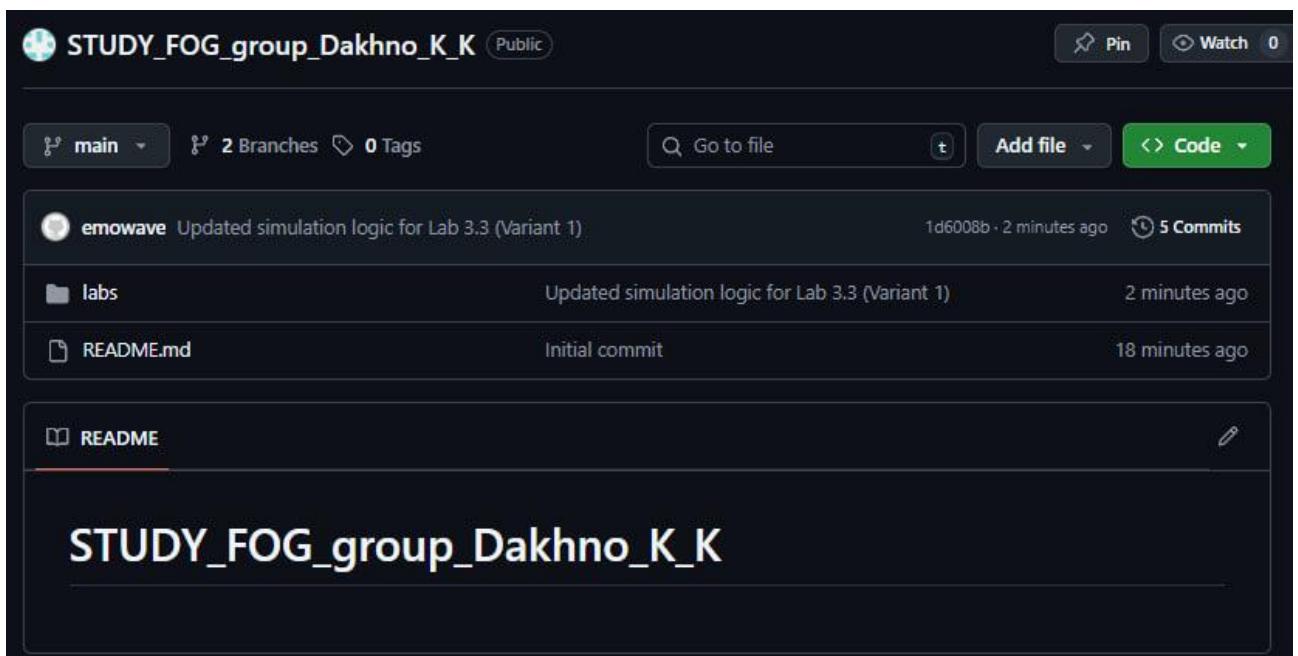


Рисунок 6 – Работа с репозиторием через терминал

Ссылка на репозиторий:

https://github.com/ayobuben/STUDY_FOG_group_Dakhno_K_K

Изм.	Лист	№ докум.	Подпись	Дата

УП.530000.000

Лист

7

Вывод: В ходе лабораторной работы были освоены ключевые принципы работы с распределенной системой контроля версий Git.

1. Приобретен навык создания и администрирования репозиториев на GitHub.
2. Изучен процесс безопасного внесения изменений через ветвление (New Branch) и запросы на слияние (Pull Request), что является стандартом в командной разработке.
3. Освоены основные консольные команды (clone, add, commit, push), позволяющие вести разработку локально и синхронизировать код с удаленным сервером.

Созданный репозиторий теперь служит единым хранилищем отчетов и программного кода по курсу.

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

УП.530000.000

Лист

7