ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ

«КОЛЛЕДЖ АВТОМАТИЗАЦИИ И ИНФОРМАЦИОННЫХ

ТЕХНОЛОГИЙ № 20»

**ОТЧЕТ   
О ПРОХОЖДЕНИИ ПРАКТИКИ**

**УЧЕБНАЯ ПРАКТИКА**

(наименование вида практики)

Тугоншоев Шохзур Тугоншоевич

(Ф. И. О. обучающегося)

**ПМ.02 Осуществление интеграции программных модулей**

курс 2

группа ИСП212д

формы обучения очная

специальность/профессия 09.02.07 Информационные системы и программирование

(код и название)

Место прохождения практики ГБПОУ КАИТ №20 г. Москва ул. 1-ая Мясниковская, д.16

Срок прохождения практики с «21» апреля 2025г. по «25» апреля 2025г.

В объеме 36 часов

Отчёт принят

25.04.2025г., преподаватель, Кузьмина О.Б. \_\_\_\_\_\_\_\_\_\_

(подпись)

25.04.2025г., преподаватель, Новиков Г.С. \_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

г. Москва

2025 г.

Содержание

[Введение 3](#_Toc196293458)

[Основная часть 4](#_Toc196293459)

[1. Постановка задачи 4](#_Toc196293460)

[2. Выбор средств разработки 4](#_Toc196293461)

[3. Структура программы 4](#_Toc196293462)

[4. Описание реализации 5](#_Toc196293463)

[5. Оформление интерфейса 5](#_Toc196293464)

[6. Тестирование программы 6](#_Toc196293465)

[7. Выводы по реализации 6](#_Toc196293466)

[Заключение 7](#_Toc196293467)

# **Введение**

В ходе прохождения учебной практики по модулю ПМ.02 “Осуществление интеграции программных модулей” был разработан программный проект на языке программирования c#. Целью работы являлась разработка современного графического пользовательского интерфейса для расчета и анализа показателей системы массового обслуживания (СМО) на примере модели работы стоматологической клиники.

Практика проходила в Государственном бюджетном профессиональном образовательном учреждении города Москвы “Колледж автоматизации и информационных технологий №20” (КАИТ 20 Диджитал) – современном образовательном учреждении, которое активно развивает цифровые компетенции студентов. Колледж предоставляет качественное образование в области информационных технологий, программирования и системной интеграции, что создает благоприятные условия для успешного освоения практических навыков и подготовки к профессиональной деятельности.

Создание программы, современного дизайна и обработкой ошибок позволило углубить знания в области интеграции программных модулей и разработки удобных интерфейсов. Проект предусматривает расчет вероятностных характеристик СМО и вывод результатов в наглядной и понятной форме.

## **Основная часть**

## **1. Постановка задачи**

Необходимо разработать приложение, позволяющее пользователю вводить параметры модели λ (интенсивность потока пациентов) и μ (интенсивность обслуживания), рассчитывать показатели работы системы и выводить результаты в визуальной форме. Требуется предусмотреть:

* Интуитивно понятный интерфейс;
* Анимацию процесса расчета;
* Обработку ошибок ввода;
* Красочное визуальное оформление.

## **2. Выбор средств разработки**

В качестве языка программирования выбран c#, благодаря его простоте и универсальности.

## **3. Структура программы**

* Программа состоит из следующих основных компонентов:
* Главное окно приложения с градиентным фоном;
* Формы для ввода данных λ и μ;
* Прогресс-бар для визуализации процесса расчета;
* Кнопка запуска расчета;
* Поле для вывода результатов расчетов;
* Стилизация элементов управления для создания современного дизайна.

(См. Рис. 1-4)

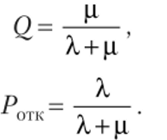
## **4. Описание реализации**

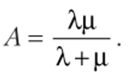
При запуске приложения пользователю предлагается ввести два параметра: интенсивность потока пациентов (λ) и интенсивность обслуживания (μ). После нажатия на кнопку "Рассчитать результаты" программа проверяет корректность введенных данных.

При корректных данных начинается анимация заполнения прогресс-бара. После завершения расчета выводятся следующие результаты:

* Вероятность отказа в обслуживании (P);
* Относительная пропускная способность системы (Q);
* Абсолютная пропускная способность (A).

Формулы расчета:





(См. Рис. 2-4)

## **5. Оформление интерфейса**

Для повышения удобства использования и улучшения визуального восприятия были реализованы:

* Градиентный фон, создающий эффект плавного перехода цвета;
* Анимированное заполнение прогресс-бара;
* Современные шрифты и цветовые схемы;
* Отступы и выравнивание элементов для комфортной работы пользователя.

**(См. Рис. 5-8)**

## **6. Тестирование программы**

Проведено тестирование программы с различными наборами входных данных, в том числе:

* Положительные действительные числа;
* Нулевые и отрицательные значения (для проверки обработки ошибок);
* Пограничные случаи с малыми значениями.

Программа корректно обрабатывает все тестовые сценарии, обеспечивает расчет необходимых показателей и вывод результатов в понятной форме.

**(См. Рис. 5-8)**

## **7. Выводы по реализации**

Разработанное приложение соответствует поставленной задаче, обладает удобным интерфейсом, реализует расчет всех необходимых показателей модели СМО и обеспечивает обработку возможных ошибок ввода. Программа может быть расширена в дальнейшем, например, для поддержки многоканальных СМО, графического отображения результатов или сохранения данных в файл.

# **Заключение**

В результате прохождения учебной практики я получил важные навыки разработки пользовательских интерфейсов с использованием языка c#. Я закрепил знания в области интеграции программных модулей, освоил методы визуализации процесса вычислений и научился применять стилизацию для улучшения пользовательского опыта.

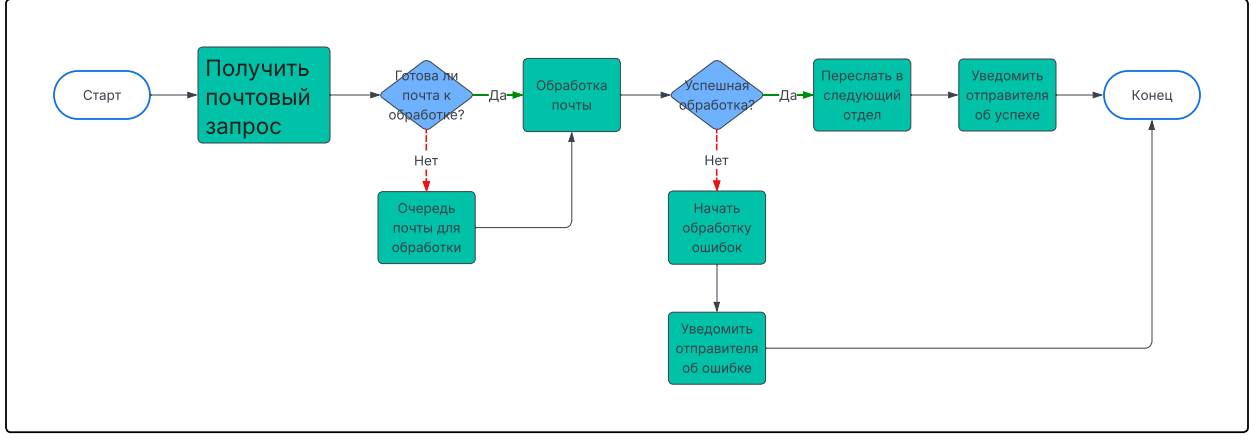
Проект продемонстрировал важность качественного оформления программных продуктов и актуальность навыков GUI-разработки. Также работа позволила мне углубить понимание математических основ систем массового обслуживания, что будет полезно в дальнейшей профессиональной деятельности.

Ссылка на репозиторий с материалами -

<https://github.com/No1lq/uch_praktika/tree/main/SMO_Efficiency_Calculator>

## Приложение 1:

Блок-схема алгоритма:

Рис.1 Блок-схема алгоритма

## Приложение 2:

Код Программы:

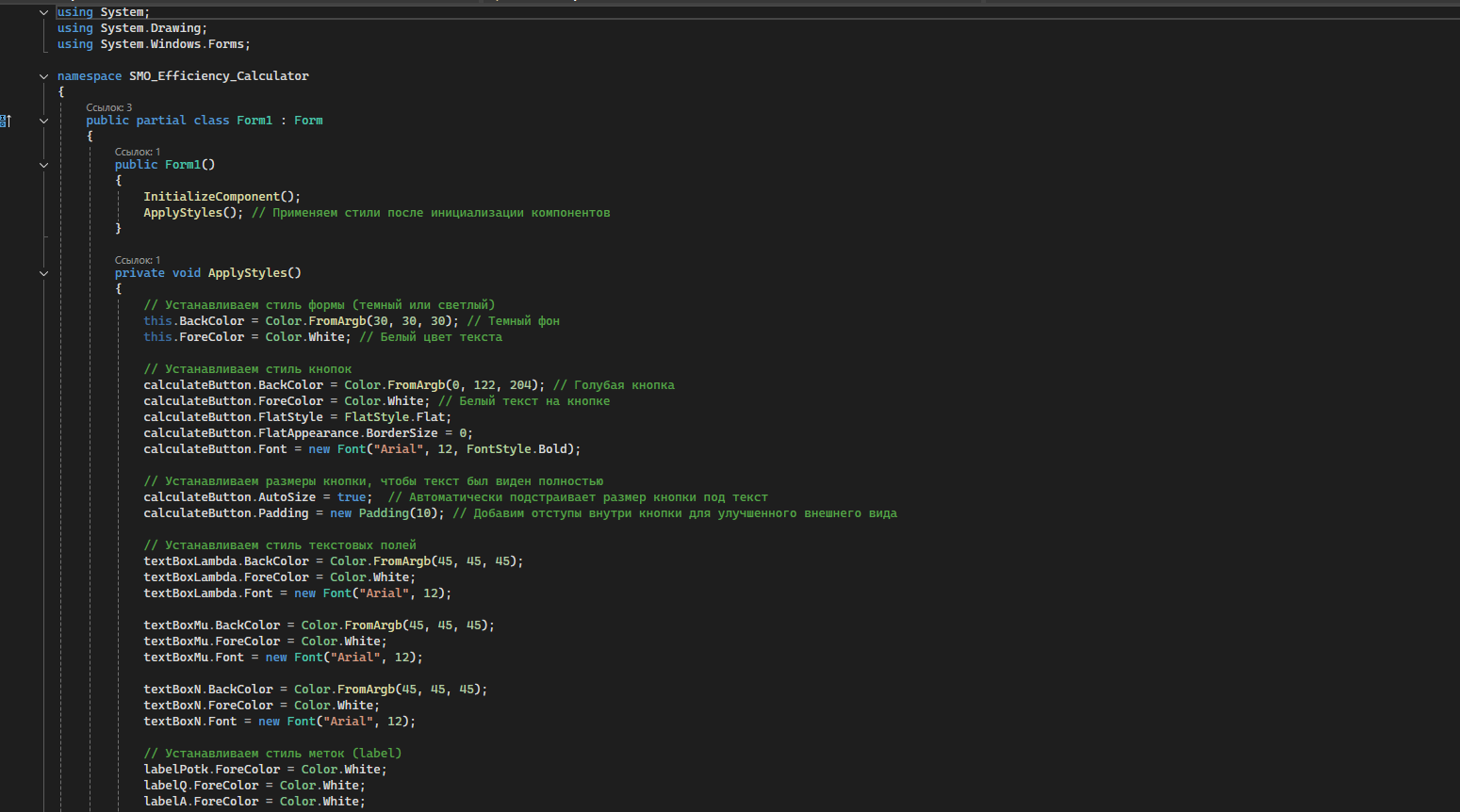


Рис. 2 Код программы

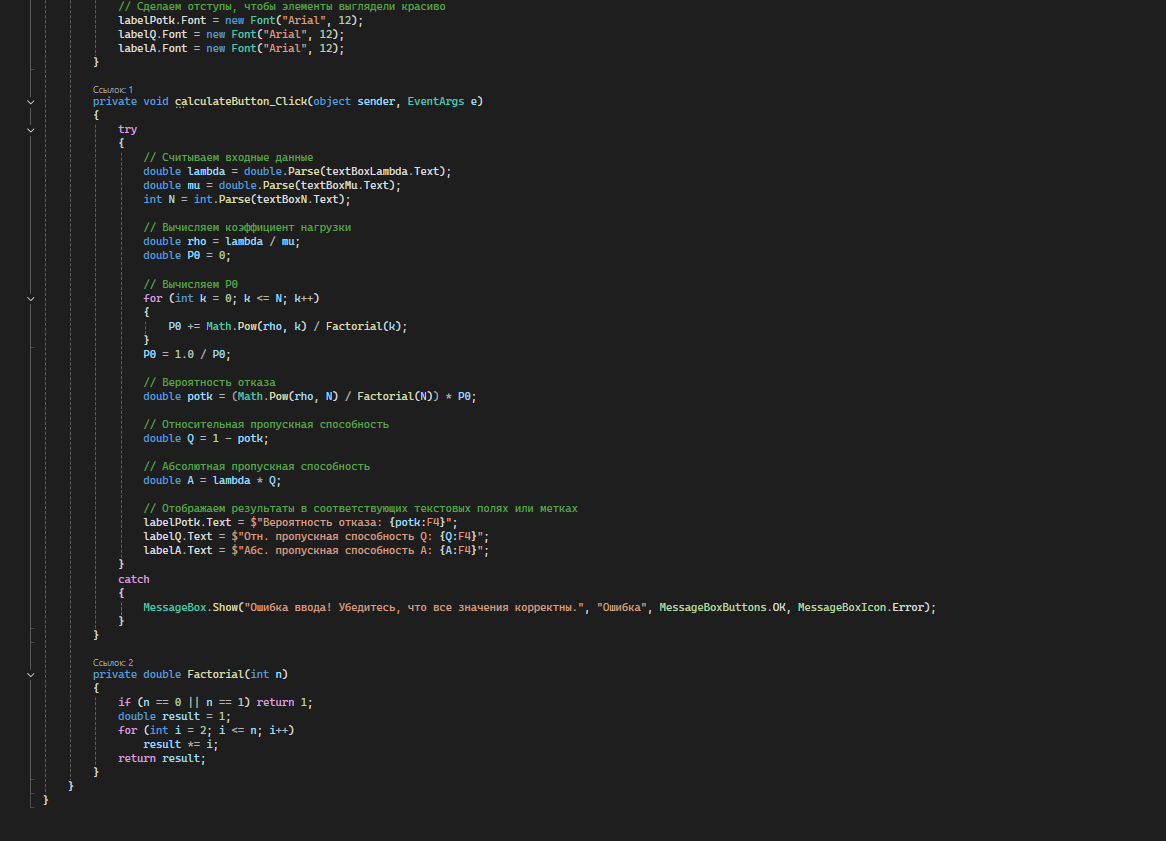


Рис. 3 Код программы

## Приложение 3:

Работа программы:

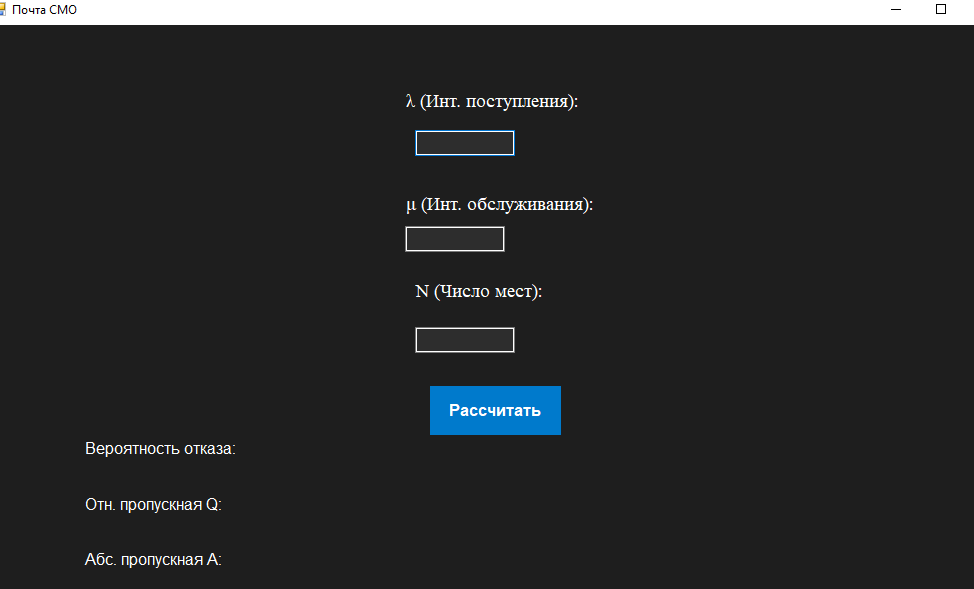


Рис. 4 Интерфейс программы

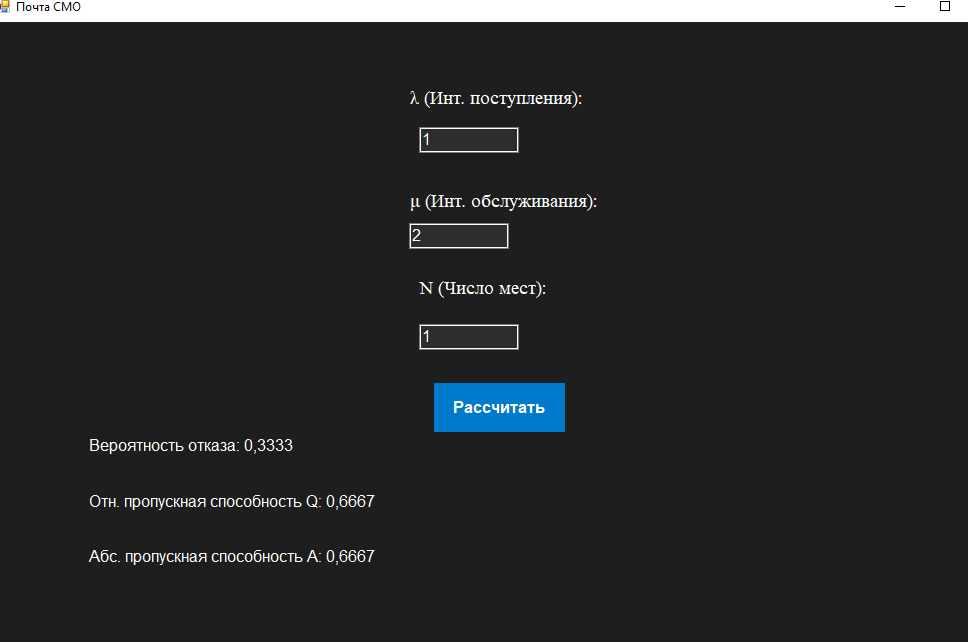


Рис. 5 Тест программы на «нормальные числа»

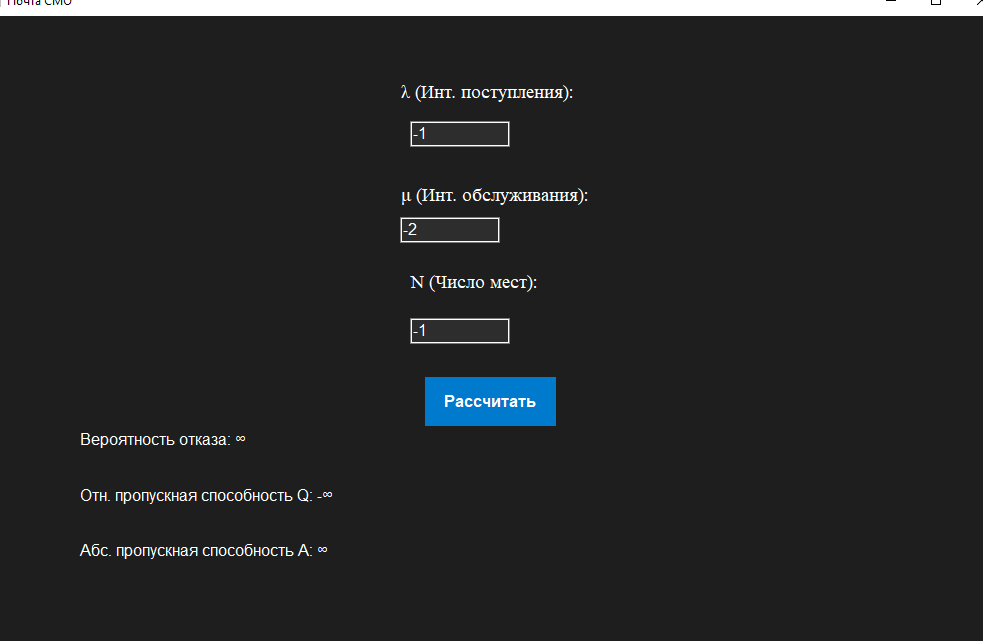


Рис. 7 Тест программы на «отрицательные числа»

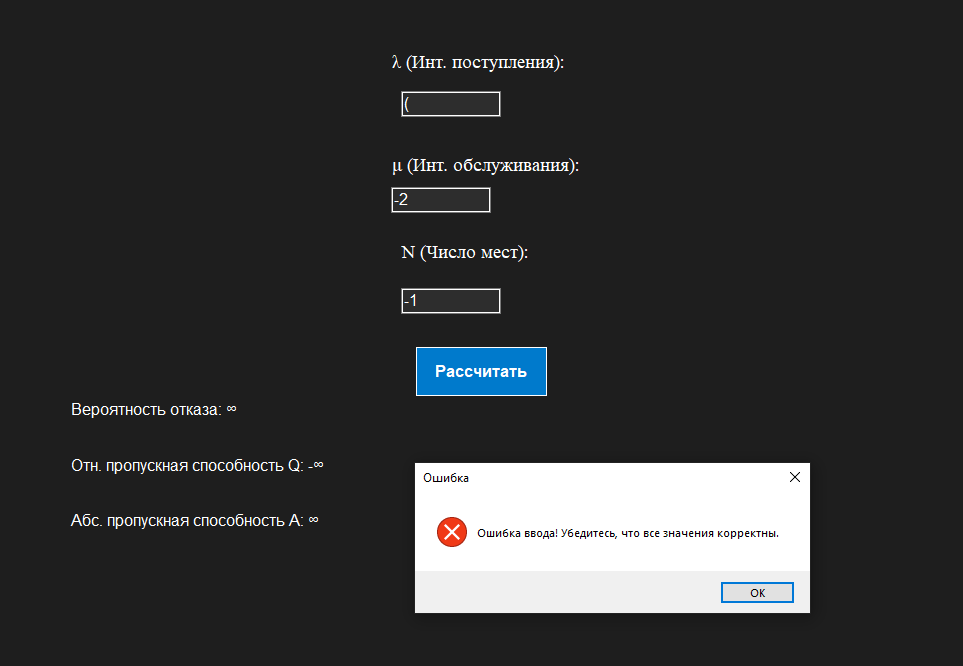


Рис. 8 Тест программы на «малые значения»

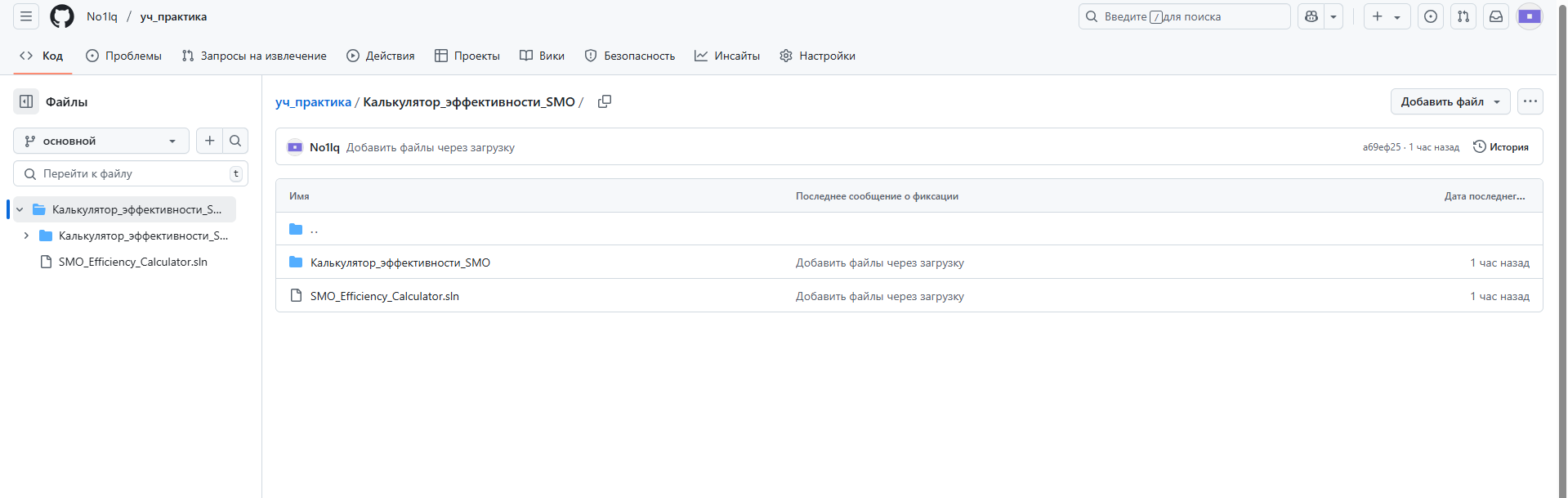


Рис. 9 Загрузка приложения на сайт Github

# **Список литературы**