МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра информационных технологий**

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3**

**по дисциплине  
  «Технологии проектирования ПО»**

Выполнил студент группы 35/2                                              \_\_\_\_           А.В. Акулов

Направление подготовки  02.03.03  Математическое обеспечение и администрирование информационных систем

Курс    3

Краснодар

2025 г.

**1. Назначение и общая цель создания программы**

Сервис сокращения ссылок предназначен для автоматизации преобразования длинных URL-адресов в короткие, удобные для использования в социальных сетях, мессенджерах и других платформах с ограничением длины сообщений.

**Цель:**

* Повышение удобства работы пользователей с длинными ссылками.
* Сбор статистики переходов по сокращенным ссылкам.
* Обеспечение безопасности и защиты от злоупотреблений.

**2. Структура программы и состав функциональных задач**

Программа состоит из следующих модулей:

1. **Веб-интерфейс для пользователей**:

ввод длинного URL;

отображение сокращенной ссылки;

просмотр статистики переходов.

1. **API для интеграции**:

генерация коротких ссылок программно.

1. **База данных**:

хранение соответствий "длинный URL — короткий код";

учет статистики переходов.

1. **Система перенаправления**:

обработка запросов по коротким ссылкам.

**3. Функциональные требования**

**3.1. Основные функции:**

* Пользователь должен иметь возможность ввести длинный URL и получить уникальную короткую ссылку.
* Система должна автоматически генерировать короткий код (например, shrt.li/abc12).
* При переходе по короткой ссылке пользователь должен быть перенаправлен на исходный URL.
* Система должна вести статистику: количество переходов, дата создания, геолокация пользователей.

**3.2. Требования к данным:**

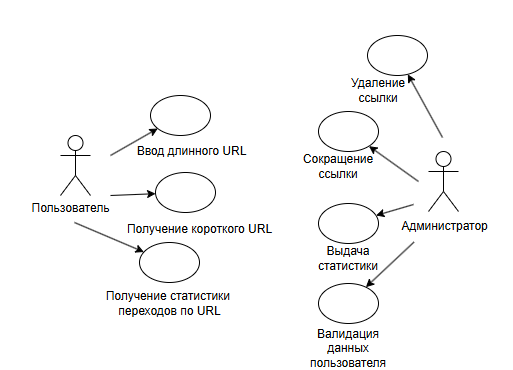
* Длинный URL должен быть валидным (проверка на соответствие формату).
* Короткий код должен быть уникальным (проверка на дублирование в БД).

**3.3. Обработка ошибок:**

При неверном вводе URL система должна выводить сообщение об ошибке.

При попытке создать дублирующую короткую ссылку система должна предложить альтернативный вариант.

**4. Модель требований UML**

**Диаграмма вариантов использования:**  
****

**Описание:**

* Актеры: Пользователь, Администратор.
* Прецеденты:
  + Создать короткую ссылку: Пользователь вводит URL → система генерирует короткий код.
  + Просмотреть статистику: Пользователь/Администратор получает данные о переходах.
  + Удалить ссылку: Администратор удаляет неактуальные записи из БД.

**5. Требования к информационному обеспечению (БД)**

* Таблицы:
  + links: id (PK), long\_url, short\_code, created\_at.
  + stats: id (PK), link\_id (FK), visit\_time, ip\_address.
* Ограничения:
  + Уникальность short\_code.
  + Автоматическое удаление записей старше 2 лет.

**6. Требования к инструментальному ПО**

* СУБД: PostgreSQL.
* Язык программирования: Python (Django).
* Средства разработки: PyCharm, Git.
* Хостинг: AWS или Heroku.

**7. Нефункциональные требования**

* Производительность: Обработка до 1000 запросов в секунду.
* Безопасность: Защита от DDoS-атак, валидация входных данных.
* Масштабируемость: Возможность горизонтального масштабирования.
* Удобство интерфейса: Минималистичный дизайн, поддержка мобильных устройств.

