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 № 7**

**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. **Описание специального ПО и его компонентов**

**Общее описание разработанного ПО:**

Разработано веб-приложение для сокращения URL-ссылок с REST API.  
Ключевые функции:

* Приём длинной ссылки и возврат короткой.
* Перенаправление по короткому коду.
* Учёт статистики (дата, количество переходов).
* Работа с пользователями (создание, привязка ссылок).

**Технологии:**

* Язык: **Go (Golang)**.
* Веб-фреймворк: **Gin**.
* ORM: **GORM**.
* База данных: **PostgreSQL**.
* Документация API: **Swagger / OpenAPI**.
* Инструменты: **Postman**, **Docker (опционально)**.

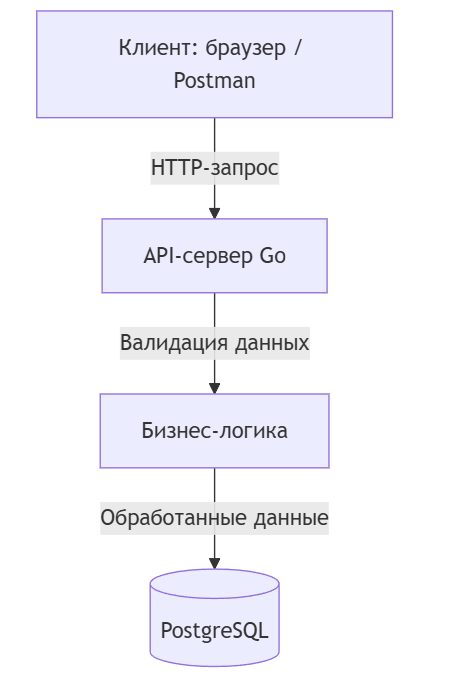
**Перечень компонентов приложения**

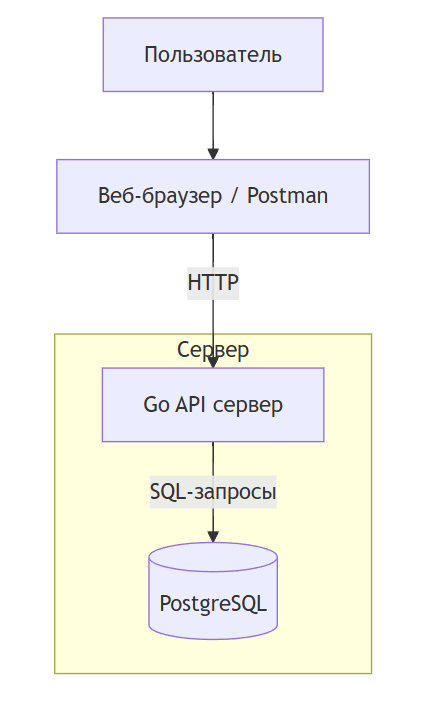
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № | |  | | --- | |  |  |  | | --- | | Имя компонента | | Тип | |  | | --- | |  |  |  | | --- | | Описание | |
| 1 | |  | | --- | |  |  |  | | --- | | main.go | | source | Точка входа. Запускает сервер и маршруты. |
| 2 | models/user.go | source | Структура пользователя (User), поля, связи |
| 3 | |  | | --- | |  |  |  | | --- | | models/link.go | | source | Модель короткой ссылки (Link), hash, user\_id. |
| 4 | |  | | --- | |  |  |  | | --- | | models/stat.go | | source | Модель статистики (Stat): дата, клики, связь. |
| 5 | |  | | --- | |  |  |  | | --- | | handlers/link.go | | source | Обработчики API: сокращение, редирект, статистика. |
| 6 | |  | | --- | |  |  |  | | --- | | db/db.go | | source | Настройка подключения к БД через GORM. |
| 7 | |  | | --- | |  |  |  | | --- | | routes.go | | source | Инициализация и регистрация маршрутов. |

**Физические элементы программной системы**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **№** | |  | | --- | |  |  |  | | --- | | **Имя узла** | | |  | | --- | |  |  |  | | --- | | **Тип** | | Описание |
| 1 | |  | | --- | |  |  |  | | --- | | go-api-server | | processor | Сервер Go-приложения, обрабатывающий HTTP-запросы. |
| 2 | postgres-db | database | Реляционная база данных PostgreSQL. |
| 3 | клиент | device | Устройство пользователя с браузером/Postman. |

1. **Диаграмма компонентов UML**



**3. Диаграмма развертывания UML**

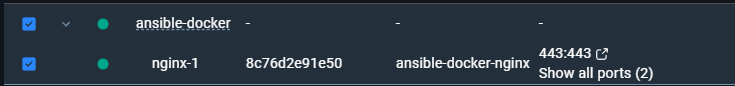
**4. Описание узлов**

|  |  |
| --- | --- |
| Узел | Описание |
| Пользовательское устройство | Браузер, Postman или иное ПО, формирующее HTTP-запрос |
| API-сервер (Go) | Backend-приложение на Go, развёрнутое на хосте / VPS |
| БД PostgreSQL | Сервер PostgreSQL, может быть локальным или удалённым |

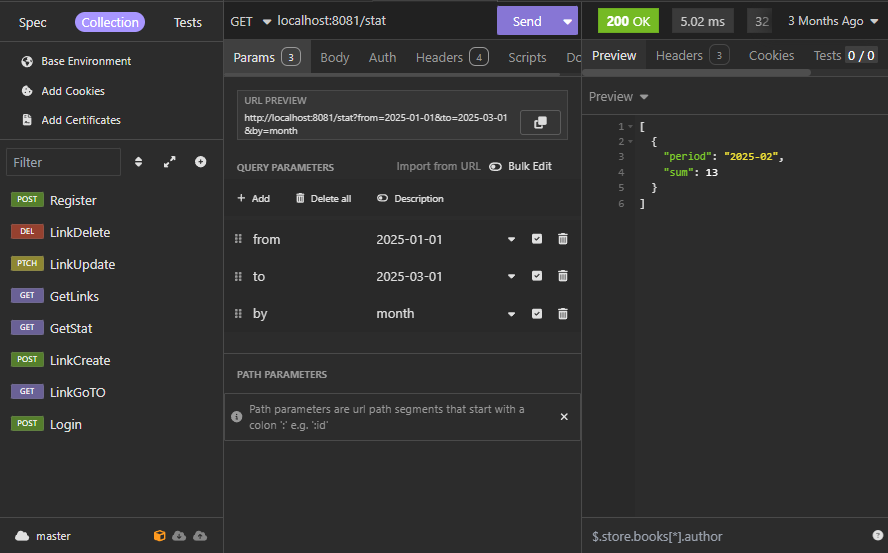
**5. Модель исполнения (использования)**

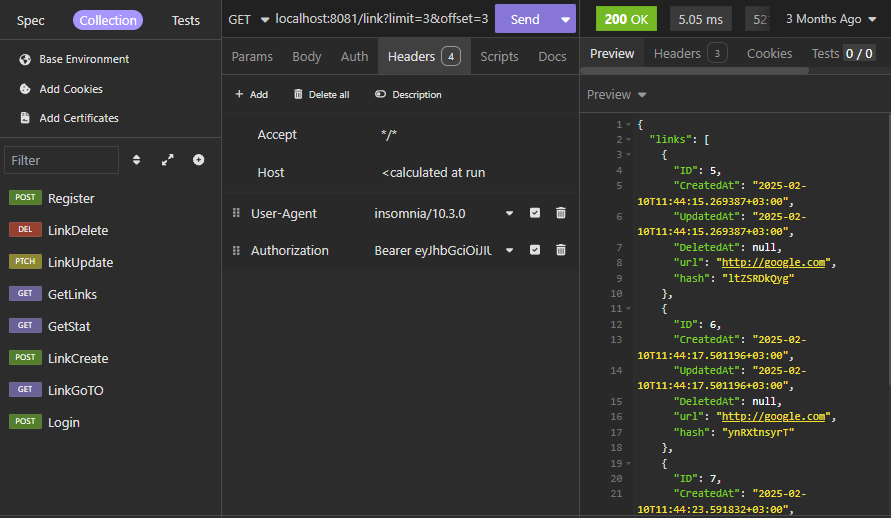
1. Пользователь вводит URL в интерфейсе или отправляет HTTP-запрос.
2. API-сервер принимает запрос и валидирует данные.
3. Сервисная часть генерирует уникальный хеш-ссылки.
4. Данные сохраняются в PostgreSQL через GORM.
5. API-сервер возвращает сгенерированную короткую ссылку клиенту.
6. **Скриншоты выполнения приложения**

Запущенные контейнеры приложения:



Успешные ответы от API:





1. **Экранные формы**

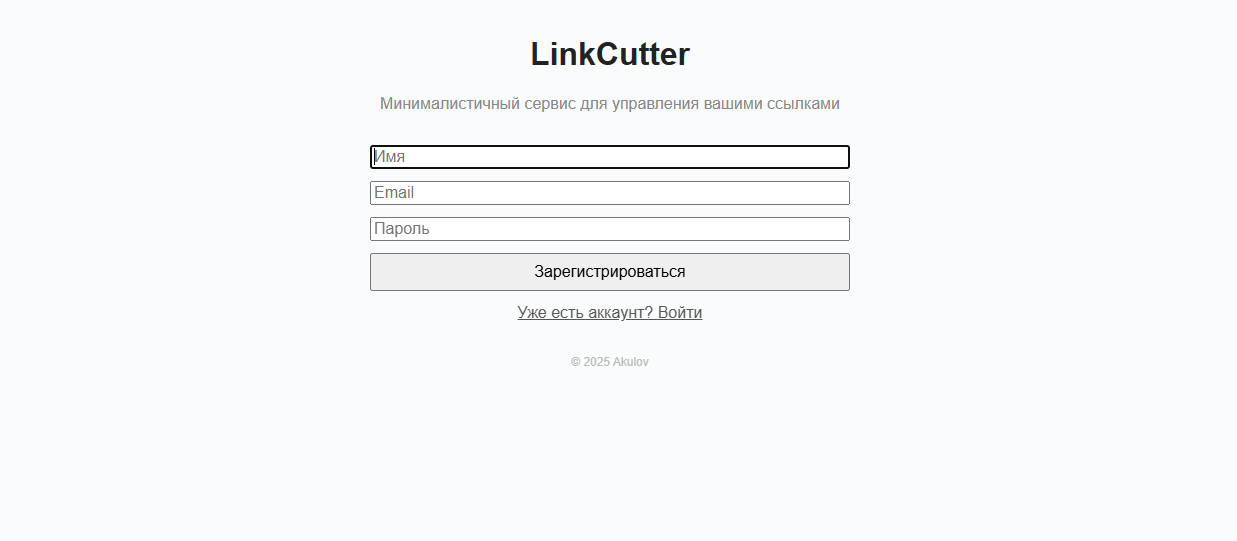


Рисунок 1 – Страница регистрации сервиса

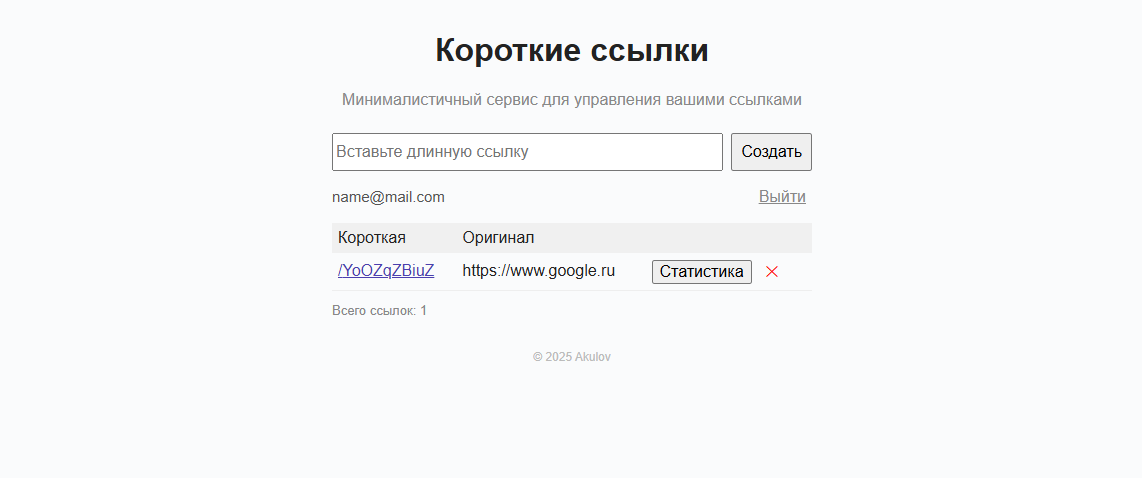


Рисунок 2 – Главная пользовательская страница сервиса

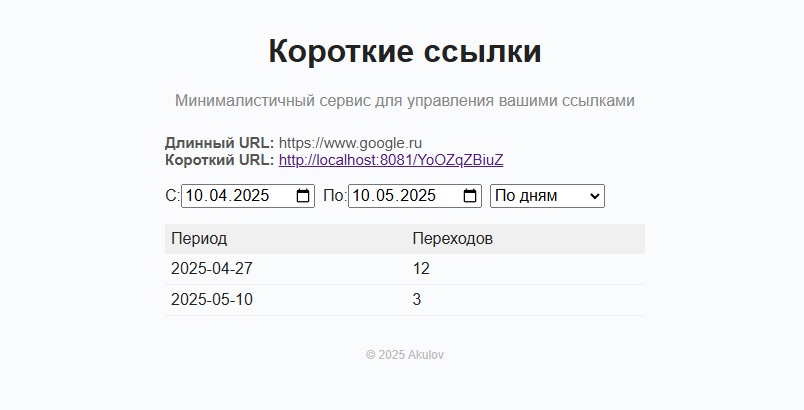


Рисунок 3 – Страница статистики

1. **Анализ сложности кода**

| **Показатель** | **Значение (пример)** |
| --- | --- |
| Количество файлов/модулей | 7 |
| Количество структур (User, Link, Stat) | 3 |
| Кол-во методов / обработчиков API | 8–10 |
| Переменные в функциях (суммарно) | ~30 |
| Операторы (if, for, select и т.п.) | ~50 |
| Глубина вложенности | До 3 уровней |
| Глубина наследования | Нет (Go не использует классы) |

1. **Выводы**

В ходе лабораторной работы спроектирована архитектура и описано развертывание микросервиса сокращения ссылок, написанного на Go. Были использованы диаграммы компонентов и развертывания UML для визуализации связей между элементами системы, а также дана подробная характеристика используемых компонентов и узлов.