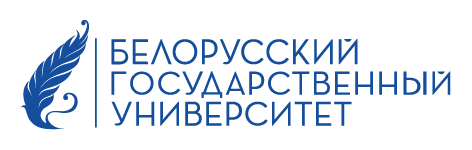
**ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ**

**Отчёт по лабораторной работе №4**

**по дисциплине “Технологии программирования для мобильных приложений”**

**CountryApp: Консольное приложение для управления данными о странах и регионах**

*Студента 2 курса 12 группы*

*Набиев Озоджон Преподаватель: Давидовская М.И.*

Минск, 2025

# ОПИСАНИЕ ПРОЕКТА

* **Цель**: Разработка консольного приложения на языке C для управления данными о странах и регионах с использованием SQLite. **Основной функционал**: Аутентификация пользователей
* Добавление, удаление и просмотр стран и регионов
* Просмотр статистики: среднее население регионов, общее население стран **Технологии**:
* Язык программирования: C
* База данных: SQLite (countries.db)
* Инструменты: Git/GitHub, Draw.io (для UML-диаграмм)

# СТРУКТУРА ПРОЕКТА

CountryApp/

├── bin/ # Исполняемые файлы ├── build/ # Объектные файлы

├── includes/ # Заголовочные файлы (.h)

│ ├── auth.h

│ ├── database.h

│ ├── country.h

│ └── region.h

├── src/ # Исходные файлы (.c)

│ ├── main.c │ ├── auth.c

│ ├── database.c

│ ├── country.c

│ └── region.c

├── diagrams/ # UML-диаграммы

├── countries.db # База данных SQLite

├── init\_db.sql # Скрипт для создания БД

├── Makefile # Файл для сборки

└── README.md # Документация

**Краткое пояснение**: Код разделён на модули: аутентификация, работа с БД, управление странами и регионами. UML-диаграммы описывают архитектуру и функциональность.

# ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

•**Аутентификация**:

•Вход пользователя по логину и паролю

•**Управление странами**:

•Добавление новой страны

•Удаление страны

•Просмотр информации о стране •**Управление регионами**:

•Добавление региона

•Удаление региона •**Просмотр данных**:

•Список регионов для выбранной страны

•Среднее население регионов страны

•Общее население всех стран

**Комментарий**: Все требования реализованы в приложении bin/country\_app.

**АРХИТЕКТУРА ПРИЛОЖЕНИЯ (UML-ДИАГРАММЫ)**

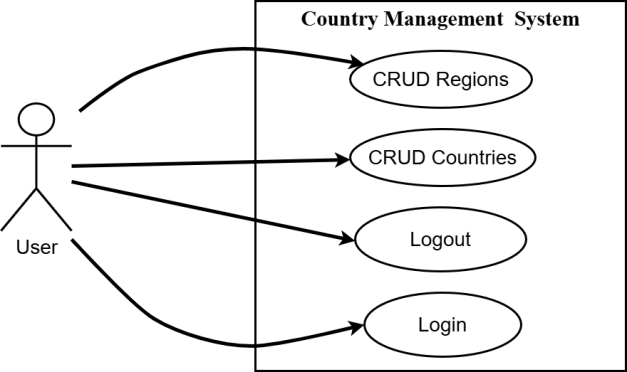
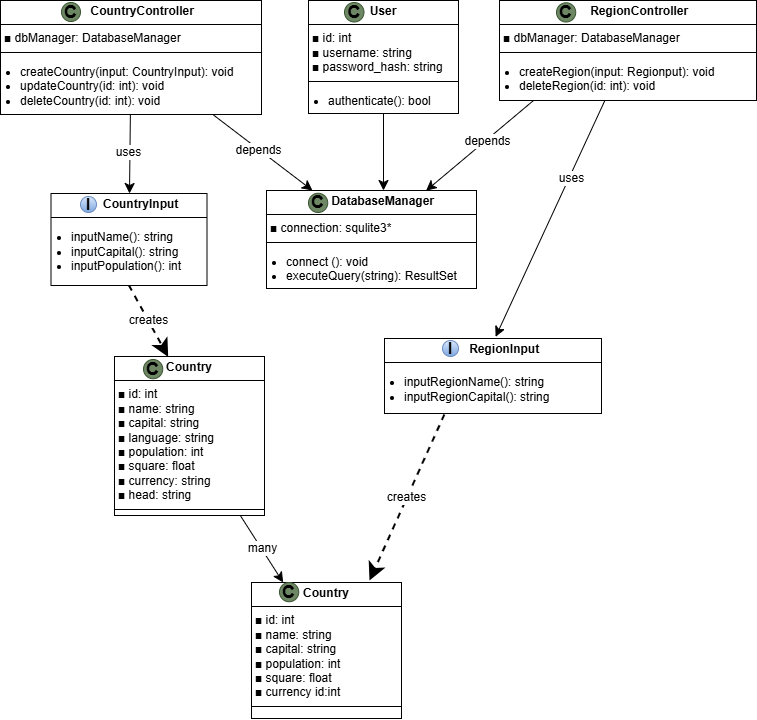
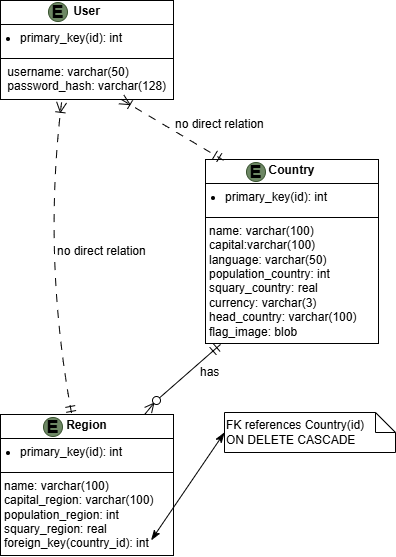
* **ER-диаграмма**: Схема базы данных

**ER-диаграмма**: 3 таблицы (country, region, user) с связью 1 ко многим между country и region.

* **Диаграмма классов:** Структура • **Диаграмма вариантов**

классов **использования:** Основные

функции



**Классы**: Сущности (Country, Region, **Варианты использования**: Основные User), интерфейсы (CountryInput, действия пользователя (вход, RegionInput), контроллеры. добавление/удаление данных, просмотр статистики).

# РЕАЛИЗАЦИЯ (КОД И ИНТЕРФЕЙС)

**Основной код** (пример функции add\_country):

void add\_country(sqlite3 \*db) { char name[100], capital[100], language[50], currency[50], head[100]; int population; float square;

printf("Enter country name: "); scanf("%s", name);

// ... (остальной ввод данных) char sql[500];

snprintf(sql, sizeof(sql),

"INSERT INTO country (name, capital, language, population\_country,

square\_country, currency, head\_country) "

"VALUES ('%s', '%s', '%s', %d, %f, '%s', '%s')", name, capital, language, population, square, currency, head); // ... (выполнение запроса) } **Интерфейс:**

CountryApp Menu:

1. Add Country
2. Delete Country
3. Add Region
4. Delete Region
5. List Regions for Country
6. Average Population by Regions
7. Total Population
8. Exit

Choose an option:

# РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ

**Тест 1: Аутентификация** Успешный вход: admin/admin123 → Доступ разрешён

* Неуспешный вход: wrong/wrong → Ошибка

**Тест 2: Добавление страны**

* Добавлена страна: Japan, Tokyo, Japanese, 125000000, 377975.0, JPY, Emperor
* Проверено в countries.db → Данные сохранены

**Тест 3: Просмотр регионов** • Для страны с ID 1 (Беларусь): Regions for country ID 1:

* Minsk Region (Capital: Minsk)
* Gomel Region (Capital: Gomel)

**Тест 4: Статистика**

* Среднее население регионов (Беларусь): 1400000.00
* Общее население стран: 1545000000

# ДОКУМЕНТИРОВАНИЕ

**README.md**:

* Описание проекта, инструкции по установке и использованию
* Ссылка: [README.md](https://github.com/fpmi-tp2025/tpmp-gr12-lab4-PermanAta777/blob/main/README.md)

**Wiki**:

* Разделы: Главная страница, Глоссарий, Функциональные требования, Спецификация, Схема БД, Презентация
* Ссылка: [CountryApp Wiki](https://github.com/fpmi-tp2025/tpmp-gr12-lab4-PermanAta777/wiki)

**Диаграммы**:

* 9 UML-диаграмм в папке diagrams/, включены в wiki

# ВЫВОДЫ

**Достижения**:

* Реализовано консольное приложение с полной функциональностью
* Настроен репозиторий на GitHub с ветками (main, feature/auth, feature/db, feature/models)
* Подготовлена документация (README, wiki, UML-диаграммы) **Проблемы**:
* Ограниченная обработка ошибок (например, SQL-инъекции)
* Отсутствие графического интерфейса **Дальнейшие шаги**:
* Добавить хеширование паролей для безопасности
* Реализовать редактирование стран и регионов
* Добавить графический интерфейс (например, с использованием GTK)

Спасибо за внимание!