

Колледж космического машиностроения и технологий

ОТЧЕТ

По учебной практике УП.02.01 по модулю ПМ.02 «Разработка и администрирование баз данных» специальности 09.02.03 Программирование в компьютерных системах

Выполнили студенты:

Волков М.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(подпись)

Ларченко М.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(подпись)

Преподаватель:

Гусятинер Л.Б.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(оценка)

Королев, 2021

Содержание отчёта

[**1. Проектирование базы данных 3**](#_Toc87581605)

[1.1. Построить ER-модель базы данных 3](#_Toc87581606)

[**2. Разработка приложения 3**](#_Toc87581608)

[2.1. Разработать таблицы и связи 3](#_Toc87581609)

[2.2. Разработать хранимые процедуры 4](#_Toc87581610)

[2.3. Разработать клиентское приложение 5](#_Toc87581611)

[**Приложения 7**](#_Toc87581612)

# Проектирование базы данных

## Построить ER-модель базы данных

## C:\Users\Lanode\AppData\Local\Microsoft\Windows\INetCache\Content.Word\dasdad.png

Рисунок 1. ER диаграмма №1

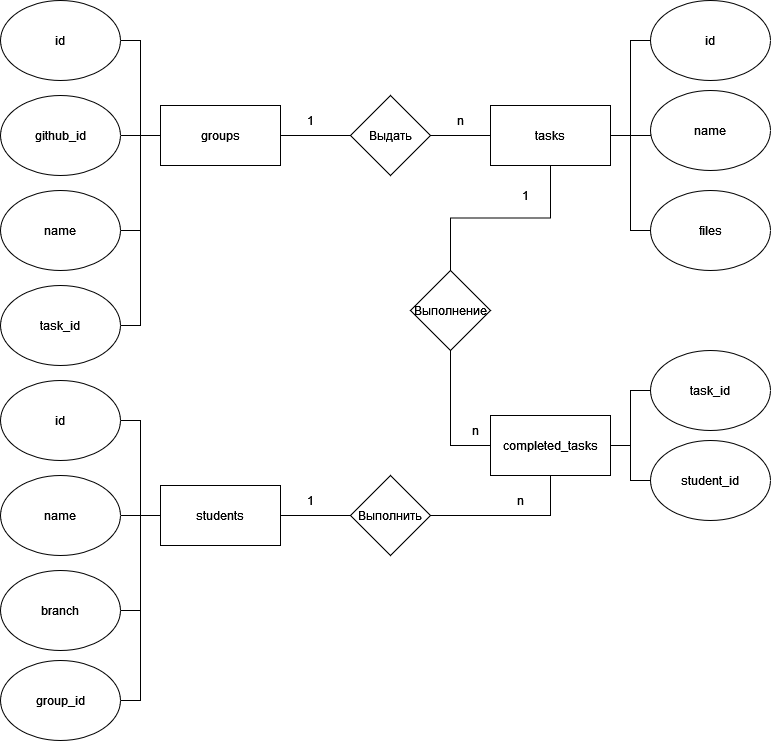


Рисунок 2. ER диаграмма №2

# Разработка приложения

## Разработать таблицы и связи

Всего разработано 4 таблицы:

* groups
* students
* tasks
* completed\_tasks

|  |
| --- |
| CREATE TABLE "tasks"  (  "id" INTEGER GENERATED ALWAYS AS IDENTITY PRIMARY KEY,  "name" varchar(64) NOT NULL,  "files" varchar(128) NOT NULL  ); |

Листинг 1. Создание таблицы заданий

|  |
| --- |
| CREATE TABLE "groups"  (  "id" INTEGER GENERATED ALWAYS AS IDENTITY PRIMARY KEY,  "github\_id" INTEGER,  "name" varchar(64) NOT NULL,  "task\_id" INTEGER,  FOREIGN KEY ("task\_id") REFERENCES "tasks" ("id") ON DELETE CASCADE  ); |

Листинг 2. Создание таблицы групп

|  |
| --- |
| CREATE TABLE "students"  (  "id" INTEGER GENERATED ALWAYS AS IDENTITY PRIMARY KEY,  "name" varchar(64) NOT NULL,  "branch" varchar(128) NOT NULL,  "group\_id" INTEGER,  FOREIGN KEY ("group\_id") REFERENCES "groups" ("id") ON DELETE CASCADE  ); |

Листинг 3. Создание таблицы студентов

|  |
| --- |
| CREATE TABLE "completed\_tasks"  (  "task\_id" INTEGER,  "student\_id" INTEGER,  FOREIGN KEY ("task\_id") REFERENCES "tasks" ("id") ON DELETE CASCADE,  FOREIGN KEY ("student\_id") REFERENCES "students" ("id") ON DELETE CASCADE  ); |

Листинг 4. Создание таблицы выполненных заданий

## Разработать хранимые процедуры

## Разработать клиентское приложение

Здесь мы можем видеть диаграмму прецедентов для приложения, на которой показаны возможные функциональные отношения (Рисунок 3).

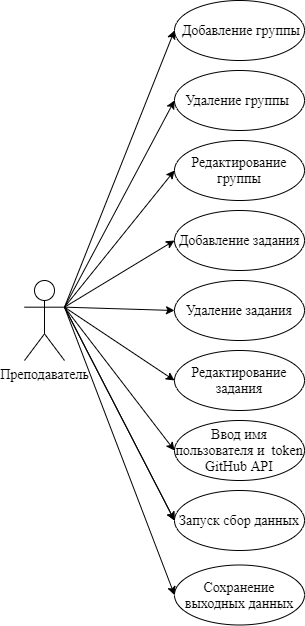


Рисунок 3. Диаграмма прецедентов приложения.

Дальше приведен сценарий использования приложения пользователем (Рисунок 4).

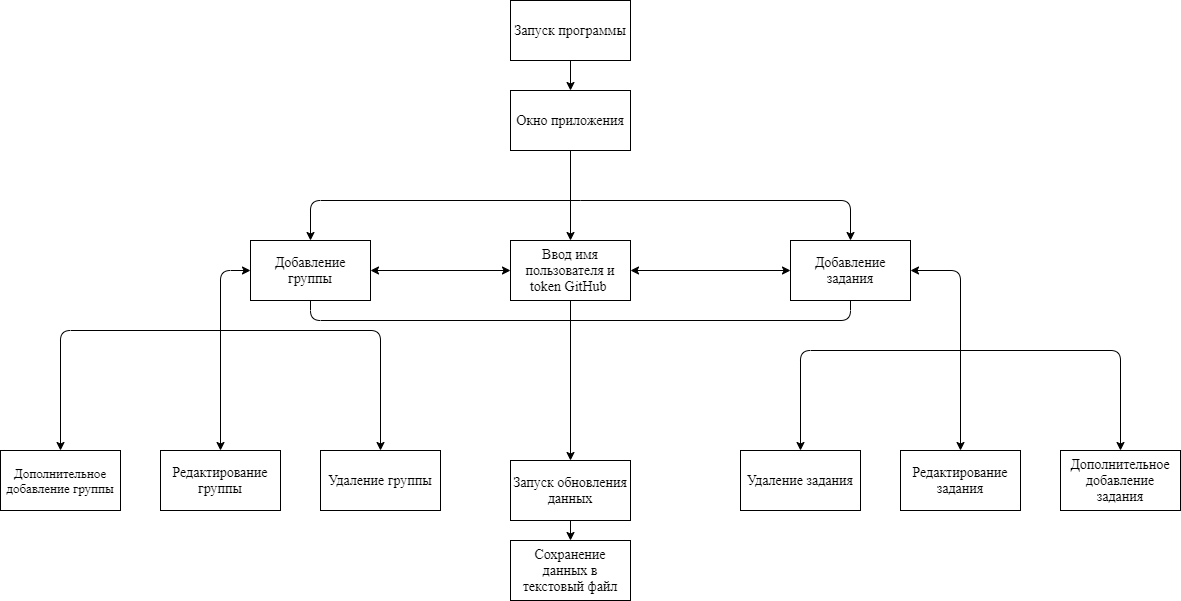


Рисунок 4. Сценарий использования приложения.

При запуске программы открывается окно приложения и у пользователя будет три действия на выбор, это: добавление группы, ввод имя пользователя и токен GitHub, добавление задания. Неважно, в каком порядке выполнять, главное сделать эти три действия, чтобы запустить обновление данных. Когда пользователь добавляет группу, появится 3 действия, это: редактирование группы, удаление группы и дополнительное добавление группы.

Когда пользователь добавляет задание, появится 3 действия, это: редактирование задания, удаление задания и дополнительное добавление задания.

После того, как пользователь обновил данные, их можно будет сохранить в текстовый файл.

# Приложения

* KKMT.Tasks.zip
* create\_db.sql
* Отчёт.docx
* Презентация.pptx