ПРИЛОЖЕНИЕ Г. скрипт базы данных

АННОТАЦИЯ

В данном программном документе представлен скрипт базы данных, структурированный следующим образом: наименование программы, область применения, скрипт базы данных.

1.1. Наименование программы: Документ содержит информацию о наименовании программы, что является ключевым идентификатором проекта.

1.2. Область применения программы: Описана область, в которой предполагается использование программы. Это важно для понимания контекста её применения.

1.3. Скрипт базы данных: Представлен скрипт базы данных, автоматически генерируемый базой данных sqlite на основе Python-кода моделей.

Данный документ служит ориентиром для разработчиков, предоставляя необходимую информацию о программе, её структуре и функциональности.

1. Скрипт базы данных
   1. Наименование программы

Наименование программы «РецептГид».

* 1. Область применения программы

Программа предназначена для использования в сфере управления рецептами, их размещении, поиске, формирования списка избранных рецептов.

* 1. Скрипт базы данных

CREATE TABLE auth\_user (

id SERIAL PRIMARY KEY,

password VARCHAR(128) NOT NULL,

last\_login TIMESTAMP WITH TIME ZONE NULL,

is\_superuser BOOLEAN NOT NULL,

username VARCHAR(150) NOT NULL UNIQUE,

first\_name VARCHAR(150) NOT NULL,

last\_name VARCHAR(150) NOT NULL,

email VARCHAR(254) NOT NULL UNIQUE,

is\_staff BOOLEAN NOT NULL,

is\_active BOOLEAN NOT NULL,

date\_joined TIMESTAMP WITH TIME ZONE NOT NULL,

description TEXT NULL,

photo VARCHAR(100) NULL,

date\_of\_birth DATE NULL,

rating FLOAT DEFAULT 0.0,

deletion\_reason TEXT NULL,

role TEXT DEFAULT 'USER'

);

CREATE TABLE recipe (

id SERIAL PRIMARY KEY,

title VARCHAR(255) NOT NULL,

user\_id INTEGER NOT NULL REFERENCES auth\_user(id) ON DELETE CASCADE,

header VARCHAR(255) NOT NULL,

cooking\_time INTEGER NOT NULL,

description TEXT NOT NULL,

photo VARCHAR(100) NOT NULL,

rating FLOAT DEFAULT 0.0,

deletion\_reason TEXT DEFAULT NULL,

servings INTEGER NOT NULL,

is\_verified BOOLEAN DEFAULT FALSE,

publication\_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT\_TIMESTAMP,

kcal\_per\_100g INTEGER NOT NULL,

protein\_per\_100g FLOAT NOT NULL,

fat\_per\_100g FLOAT NOT NULL,

carbohydrates\_per\_100g FLOAT NOT NULL

);

CREATE TABLE ingredient (

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL,

kcal\_per\_100g INTEGER NOT NULL,

protein\_per\_100g FLOAT NOT NULL,

fat\_per\_100g FLOAT NOT NULL,

carbohydrates\_per\_100g FLOAT NOT NULL

);

CREATE TABLE recipe\_step (

id SERIAL PRIMARY KEY,

recipe\_id INTEGER NOT NULL REFERENCES recipe(id) ON DELETE CASCADE,

step\_number INTEGER NOT NULL,

photo VARCHAR(100) NULL,

description TEXT NOT NULL

);

CREATE TABLE recipe\_ingredient (

id SERIAL PRIMARY KEY,

ingredient\_id INTEGER NOT NULL REFERENCES ingredient(id) ON DELETE CASCADE,

recipe\_id INTEGER NOT NULL REFERENCES recipe(id) ON DELETE CASCADE,

quantity\_in\_grams INTEGER NOT NULL

);

CREATE TABLE tag (

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL UNIQUE,

description TEXT NOT NULL

);

CREATE TABLE recipe\_tag (

id SERIAL PRIMARY KEY,

tag\_id INTEGER NOT NULL REFERENCES tag(id) ON DELETE CASCADE,

recipe\_id INTEGER NOT NULL REFERENCES recipe(id) ON DELETE CASCADE

);

CREATE TABLE message (

id SERIAL PRIMARY KEY,

sender\_id INTEGER NOT NULL REFERENCES auth\_user(id) ON DELETE CASCADE,

recipient\_id INTEGER NOT NULL REFERENCES auth\_user(id) ON DELETE CASCADE,

text TEXT NOT NULL,

timestamp TIMESTAMP WITH TIME ZONE DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE favorite\_recipe (

id SERIAL PRIMARY KEY,

recipe\_id INTEGER NOT NULL REFERENCES recipe(id) ON DELETE CASCADE,

user\_id INTEGER NOT NULL REFERENCES auth\_user(id) ON DELETE CASCADE

);

CREATE TABLE review (

id SERIAL PRIMARY KEY,

user\_id INTEGER NOT NULL REFERENCES auth\_user(id) ON DELETE CASCADE,

rating INTEGER CHECK (rating >= 1 AND rating <= 5),

title VARCHAR(255) NOT NULL,

text TEXT NOT NULL,

date\_written TIMESTAMP WITH TIME ZONE DEFAULT CURRENT\_TIMESTAMP,

deletion\_reason TEXT DEFAULT ''

);

CREATE TABLE friend (

id SERIAL PRIMARY KEY,

requester\_id INTEGER NOT NULL REFERENCES auth\_user(id) ON DELETE CASCADE,

accepter\_id INTEGER NOT NULL REFERENCES auth\_user(id) ON DELETE CASCADE

);

CREATE TABLE recipe\_review (

id SERIAL PRIMARY KEY,

review\_id INTEGER NOT NULL REFERENCES review(id) ON DELETE CASCADE,

recipe\_id INTEGER NOT NULL REFERENCES recipe(id) ON DELETE CASCADE

);