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, написанный на языке PostgreSql для хранения данных приложения.

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

СОДЕРЖАНИЕ

[1. Скрипт базы данных 3](#_Toc198136205)

[1.1. Наименование программы 3](#_Toc198136206)

[1.2. Область применения программы 3](#_Toc198136207)

[1.3. Скрипт базы данных 3](#_Toc198136208)

[1.4. Словарь данных 5](#_Toc198136209)

1. Скрипт базы данных
   1. Наименование программы

Наименование программы «DeviceMonitor».

* 1. Область применения программы

Приложение «DeviceMonitor» предоставляет функционал для контроля аварийных ситуаций устройств. Приложение включает в себя удобный интерфейс для добавления и отслеживания устройств, возможность подключать счетчик по сетевому протоколу, а также запускать сервер и подключаться к нему.

* 1. Скрипт базы данных

--Create-блок

--Таблица ролей пользователей

create table roles(

id serial primary key,

name varchar(50) not null --Название роли

);

--Таблица аутентификационных данных пользователей

create table users(

id serial primary key,

username varchar(50) not null,

password varchar(50) not null,

role\_id int references roles(id) on delete cascade --Роль пользователя (Администратор, Пользователь)

);

--Таблица расположений устройств

create table placements(

id serial primary key,

name varchar(50) not null --Название расположения

);

--Таблица устройств с физическими значениями

create table devices(

id serial primary key,

name varchar(70) not null,

turned boolean default true, --Включен ли прибор

voltage int not null, --Напряжение (с минимальным и максимальным значением)

voltage\_max int not null,

voltage\_min int not null,

amperage int not null, --Ток (с максимальным значением)

amperage\_max int not null,

temperature int not null, --Температура (с максимальным значением)

temperature\_max int not null,

capacity int not null, --Емкость

placement\_id int references placements(id)

);

--Таблица истории аварийных случаев (Если в БД появляется устройство с хотя бы одним значением, ниже минимального или выше максимального, то в таблицу добавляется запись со ссылкой на устройство и датой аварии)

create table device\_emergency\_history(

id serial primary key,

name varchar(70) not null,

turned boolean default true, --Включен ли прибор

voltage int not null, --Напряжение (с минимальным и максимальным значением)

voltage\_max int not null,

voltage\_min int not null,

amperage int not null, --Ток (с максимальным значением)

amperage\_max int not null,

temperature int not null, --Температура (с максимальным значением)

temperature\_max int not null,

capacity int not null, --Емкость

placement\_id int references placements(id),

creation\_time timestamp default CURRENT\_TIMESTAMP --Автоматическая запись текущей даты и времени

);

--Function-блок

--Функция проверки значений устройства перед добавлением или изменением, и добавление в таблицу истории аварий записи при неудовлетворении условий минимальных или максимальных значений

create or replace function check\_value\_trigger()

returns trigger as $$

begin

if NEW.voltage > NEW.voltage\_max or NEW.voltage < NEW.voltage\_min or NEW.amperage > NEW.amperage\_max or NEW.temperature > NEW.temperature\_max then

insert into device\_emergency\_history (device\_id) values (NEW.id);

end if;

return new;

end;

$$ LANGUAGE plpgsql;

--Trigger-блок

create trigger check\_value\_trigger after insert or update on devices for each row execute function check\_value\_trigger();

drop trigger check\_value\_trigger on devices

--Insert-блок

insert into roles (name) values ('Администратор');

insert into roles (name) values ('Пользователь');

insert into users (username, password, role\_id) values ('admin', 'admin1', 1);

insert into users (username, password, role\_id) values ('vasya123', '123', 2);

insert into placements(name) values ('Альфа');

insert into placements(name) values ('Бета');

insert into devices(name, turned, voltage, voltage\_max, voltage\_min, amperage, amperage\_max, temperature, temperature\_max, capacity, placement\_id) values ('ШИБП 1-1', true, 253, 231, 209, 36, 40, 52, 60, 150, 1);

insert into devices(name, turned, voltage, voltage\_max, voltage\_min, amperage, amperage\_max, temperature, temperature\_max, capacity, placement\_id) values ('ШИБП 1-2', true, 220, 231, 209, 32, 40, 44, 60, 150, 2);

insert into devices(name, turned, voltage, voltage\_max, voltage\_min, amperage, amperage\_max, temperature, temperature\_max, capacity, placement\_id) values ('ЩМ-120', true, 193, 231, 209, 44, 40, 51, 60, 210, 1);

--Select-блок

select \* from roles;

select \* from users;

select \* from devices;

select \* from placements;

select \* from device\_emergency\_history;

--Вывод всех устройств, находящиеся в текущий момент в аварийной ситуации

select 'Аварийная ситуация' as Статус, \* from devices d join placements p on d.placement\_id = p.id where d.voltage > d.voltage\_max or d.voltage < d.voltage\_min or d.amperage > d.amperage\_max or d.temperature > d.temperature\_max;

--Copy-блок

--Сохранение данных всех таблиц в отдельный файл формата csv

copy roles to '/tmp/roles.csv' with (format csv, header);

copy users to '/tmp/users.csv' with (format csv, header);

copy devices to '/tmp/devices.csv' with (format csv, header);

copy placements to '/tmp/placements.csv' with (format csv, header);

copy device\_emergency\_history to '/tmp/device\_emergency\_history.csv' with (format csv, header);

--Сохранение данных о всех аварийных устройствах в отдельный файл формата csv

copy (select 'Аварийная ситуация' as Статус, \* from devices d join placements p on d.placement\_id = p.id where d.voltage > d.voltage\_max or d.voltage < d.voltage\_min or d.amperage > d.amperage\_max or d.temperature > d.temperature\_max) to '/tmp/danger\_devices.csv' with (format csv, header);

--Drop-блок

drop table devices\_placements;

drop table devices;

drop table device\_emergency\_history;

drop table placements;

drop table roles;

* 1. Словарь данных

В Таблице 1 представлен словарь данных для разработанной базы

данных, с которой взаимодействует программный комплекс «DeviceMonitor».

Таблица 1 – Словарь данных

| № | Ключ | Наименование | Тип данных | Обязательность заполнения | Назначение |
| --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Таблица - roles | | | | | |
| 1 | PK | id | serial | Not null | Уникальный идентификатор |
| 2 |  | name | varchar(70) | Not null | Наименование |
| Таблица - users | | | | | |
| 3 | PK | id | serial | Not null | Уникальный идентификатор |
| 4 |  | username | varchar(50) | Not null | Логин |
| 5 |  | password | varchar(50) | Not null | Пароль |
| 6 | FK | role\_id | int | Not null | Ссылка на роль |
| Таблица - placements | | | | | |
| 7 | PK | id | serial | Not null | Уникальный идентификатор |
| 8 |  | name | varchar(50) | Not null | Наименование |
| Таблица - devices | | | | | |
| 9 | PK | id | serial | Not null | Уникальный идентификатор |
| 10 |  | name | varchar(70) | Not null | Наименование |
| 11 |  | voltage | int | Not null | Напряжение |
| 12 |  | voltage\_min | int | Not null | Минимальное напряжение |
| 13 |  | voltage\_max | int | Not null | Максимальное напряжение |
| 14 |  | amperage | int | Not null | Ток |
| 15 |  | amperage\_max | int | Not null | Максимальный ток |
| 16 |  | temperature | int | Not null | Температура |
| 17 |  | temperature\_max | int | Not null | Максимальная температура |
| 18 |  | capacity | int | Not null | Емкость |
| 19 | FK | placement\_id | int | Not null | Ссылка на расположение |
| 20 |  | turned | boolean | Not null | Включено ли устройство |
| Таблица - device\_emergency\_history | | | | | |
| 21 | PK | id | serial | Not null | Уникальный идентификатор |
| 22 |  | name | varchar(70) | Not null | Наименование |
| 23 |  | voltage | int | Not null | Напряжение |
| 24 |  | voltage\_min | int | Not null | Минимальное напряжение |
| 25 |  | voltage\_max | int | Not null | Максимальное напряжение |
| 26 |  | amperage | int | Not null | Ток |
| 27 |  | amperage\_max | int | Not null | Максимальный ток |
| 28 |  | temperature | int | Not null | Температура |
| 29 |  | temperature\_max | int | Not null | Максимальная температура |
| 30 |  | capacity | int | Not null | Емкость |
| 31 |  | placement\_id | int | Not null | Ссылка на расположение |
| 32 |  | turned | boolean | Not null | Включено ли устройство |
| 33 |  | creation\_time | timestamp | default -CURRENT\_TIMESTAMP | Время аварии |
|  |  |  |  |  |  |