Проект “Создание БД модерация сайта”

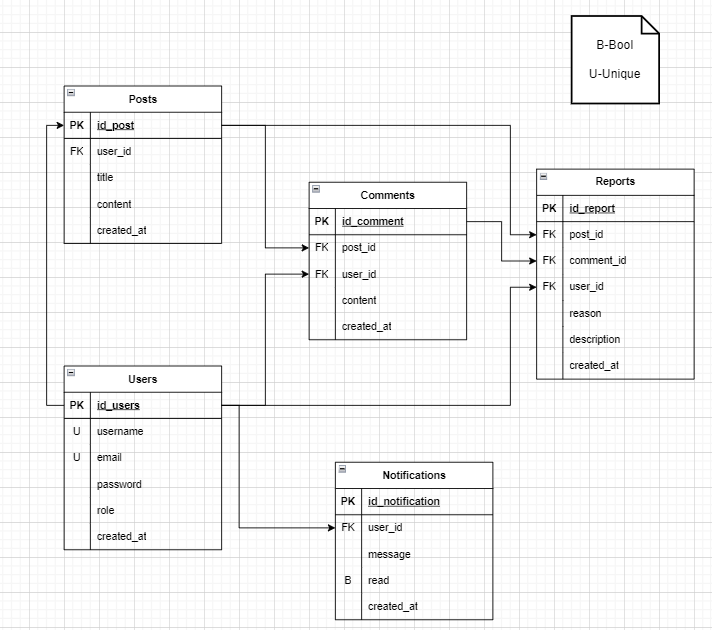
Задача: осуществления контроля и хранения сообщений и постов на сайте-мессенджере.

БД должна осуществлять:

* Аутентификация и авторизация пользователей
* Управление постами и комментариями
* Отчеты о нарушениях
* Уведомления

Разработка

**ER-модель:**



База данных содержит таблицы:

* Пользователей
* Постов от пользователей
* Комментариев от пользователей под постами
* Репортов от пользователей на комментарии или посты
* Уведомлений к пользователям

Код создания таблиц:

CREATE TABLE users (

Id\_user INT PRIMARY KEY AUTO\_INCREMENT,

username VARCHAR(50) UNIQUE NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

role ENUM('admin', 'moderator', 'user') NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE posts (

Id\_post INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

title VARCHAR(255),

content TEXT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id\_user)

);

CREATE TABLE comments (

Id\_comment INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

post\_id INT,

content TEXT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id\_user),

FOREIGN KEY (post\_id) REFERENCES posts(id\_post)

);

CREATE TABLE reports (

id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

post\_id INT,

comment\_id INT,

reason ENUM('spam', aggression', 'other'),

description TEXT,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id\_user),

FOREIGN KEY (post\_id) REFERENCES posts(id\_post),

FOREIGN KEY (comment\_id) REFERENCES comments(id\_comment)

);

CREATE TABLE notifications (

id INT PRIMARY KEY AUTO\_INCREMENT,

user\_id INT,

message TEXT,

is\_read TINYINT(1) SET DEFAULT 0,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (user\_id) REFERENCES users(id\_user)

);

Заполнение таблиц:

INSERT INTO users (username, email, password, role)

VALUES

('user1', 'user1@example.com', 'password1', 'user'),

('user2', 'user2@example.com', 'password2', 'user'),

('admin1', 'admin1@example.com', 'adm\_password1', 'admin'),

('moder1', 'moder1@example.com', mdr\_'password1', 'moderator'),

('user3', 'user3@example.com', 'password3', 'user');

INSERT INTO posts (user\_id, title, content)

VALUES (1, 'Database problems', 'I dont understand how to do something with the table I created. Any help?'),

(1, 'Database solution found', 'Okay, I found my answer, thank you all for support'),

(4, 'Stop chainposting', 'New rule for everyone: dont do posts that relate to other your posts');

INSERT INTO comments (user\_id, post\_id, content)

VALUES (2, 1, 'You need to do this thing to achieve that result'),

(1, 1, 'OMG, thank you very much, it worked'),

(2, 3, 'Dont make post about your past post, use comments for that');

INSERT INTO reports (user\_id, post\_id, comment\_id, reason, description)

VALUES (3, 2, NULL, 'other', 'Please do something with the chainposting, Im too busy rn');

INSERT INTO Notifications (user\_id, message)

VALUES (4, 'Post repoted, your action required');

Код функций:

//создание нового пользователя

create procedure new\_user(

in user\_nickname varchar(50),

in user\_email varchar(100),

in user\_password varchar(255))

begin

insert into users(username, email, password, role)

values(user\_nickname, user\_email, user\_password, 'user');

end//

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

//повышение пользователя до роли модератора(блокировка пользователя очень похожа на эту)

create procedure new\_moder(

in u\_id int)

begin

update users

set role = ‘moderator’

where id\_user = u\_id;

end//

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

//создание нового поста

create procedure new\_post(

in u\_id int,

in post\_title varchar(255),

in post\_content text)

begin

insert into posts(user\_id, title, content)

values(u\_id, post\_title, post\_content);

end//

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

//Я бы мог сделать ещё процедуры создания комментариев, репортов и уведомлений но они все одинаковые.

//Запрос на все посты от выбранного пользователя

create procedure GetUserPosts(

in u\_id int)

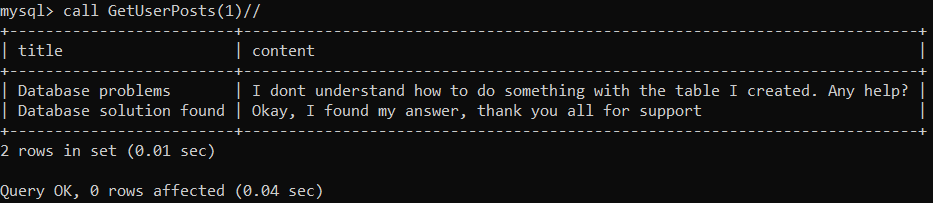
begin

select title, content

from posts

where user\_id = u\_id;

end//



//можно создать ещё очень много разных запросов, но они все будут очень похожими друг на друга.