Name :- O. Sai Nikhil DBMS :- CSA0593

Reg No:- 192325124

ASSIGNMENT - 1

Creating a database design for a social media platform with data privacy and role management is a complex task. Design a database for a social media platform where users can post follow others. and interact with posts.

1. Conceptual ERD Model:-

The conceptual ERD represents high-level entities and their relationships without specifying detailed attributes or data types.

* user : users on the platform
* post : Content created by users
* comment : comments on posts
* like : Records of likes on posts
* follow : Tracks following relationships
* Role : Defines the role of a user
* userprivacy Setting : privacy setting for u to control who can view their posts

Relation ships :-

* A user can post multiple creates
* A user can have multiple comments
* A user can like multiple posts
* A user can follow multiple other users.
* A user has one role.
* A

2. logical ERD model :-

The logical ERD includes attributes, data types and primary and foreign keys. It is a more detailed model that considers the structure of tables.

Attributes:-

1. user
* user_id (PK, INT)
* username (VARCHAR)
* email (VARCHAR)
* role_id (FK, INT, reference role)
* created_at (DATETIME)

2. Role
* role_id (PK, INT)
* role_name (VARCHAR)

3. post
* post_id (PK, INT)
* user_id (FK, INT)
* content (TEXT)
* visibility (ENUM - public, Followers)
* created_at (Datetime)

4. comment
* comment_id (PK, INT)
* post_id (FK, INT, references)
* content (TEXT)

5. like

- like_id (PK, INT)
- post_id (FK, INT, references post)
- user_id (FK, INT, references user)

6. Follow

- Follower_id (FK, INT, references user)
- setting_type (ENUM - postvisibility)
- value (Enum - public, Followers, private)

3. physical ERD model :-

The physical ERD includes detailed data types specific to the SQL database engine. This model is directly implementable.

Table definitions

```
CREATE TABLE user(
        user_id INT primary key
        username VARCHAR (50) NOT FULL,
        email VARCHAR (100) NOT FULL,
        role_id INT NOT NULL
        created_at DATETIME Default
        foreign key (role_id) references role (role_id)
        );
```

Role Table :

```
CREATE TABLE Role (
    role_id INT primary key,
    role_name VARCHAR (50) NOT Full
);
```

post table :

```
CREATE Table post (
    post_id INT primary key,
    user_id INT NOT NULL;
    content Text NOT NULL,
    visibility enum ('public', 'Followers', 'private')
    created_at Datetime Default current,
    foreign key (user_id) References user (user_id)
);
```

comment table :-

```
create Table post (
    Comment_id INT primary key
    post_id INT NOT NULL,
    user_id INT Not Null,
    content NOT NULL;
    created_at Datetime Default current_time
    foreign key (post_id) references post (post_id),
);
```

like table

```
create table like (
    like_id   INT   primary key
    post_id   INT   NOT NULL;
    user_id   INT   NOT NULL,
    created_at datetime default current_time stamp,
    foreign key (post_id) references post (post_id)
    foreign key (user_id) references user (user_id)
```

Follow table

```
create table follow (
    follower_id INT NOT NULL,
    following_id  INT  NOT NULL,
    created_at datetime default current_time stamp
  Foreign key (follower_id) references user (user_id)
  Foreign key (follower_id) references user (user_id)
```