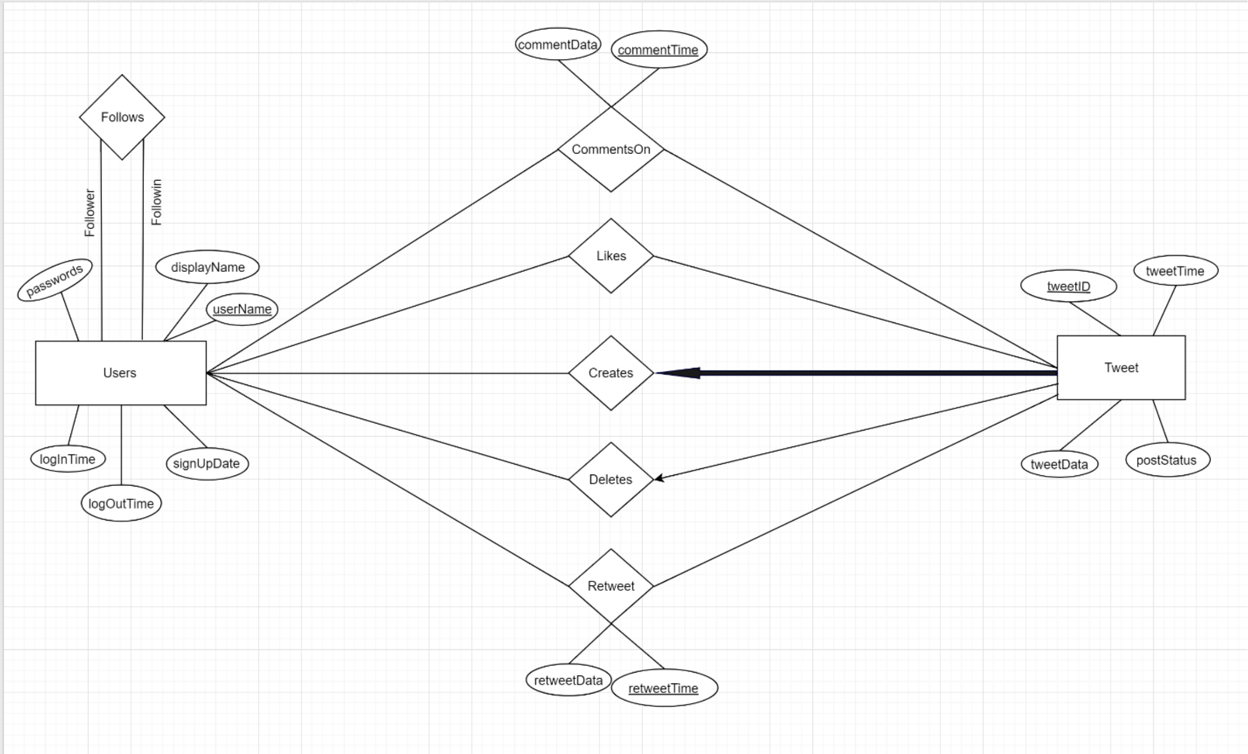
Name: Manvijay

UBIT: manvijay

UB Person Number: 50200437

Project Report CSE 460



E/R Model

- This ER diagram has two Entity sets:
    1. Users
    2. Tweet
- These two Entity sets are connected by 5 relations:
    1. CommentsOn
    2. Likes
    3. Creates (Participation constraint 1:N from Users to Tweet)
    4. Deletes (1:N relation from Users to Tweet)
    5. Retweet
- There is a relationship between the entities of the same entity set Users called Follows.

- All the attributes of these entities and relationships are self-explanatory and will be explained in the following section.

Relational Database Schema

- For every Entity set and Relationship I have created a table in the Relational Database.
- All the attributes of the Entity Sets Users and Tweets become columns in the tables with the same names.
- In all the tables I created for relationships I have added primary key attributes of the entities they are connecting besides the attributes of the relationship itself.
- For “Follows” we have two kind of User entities called Follower and Followin where Followers represents Users who are following the User and Followin represents the people who are being followed by the User.

What I did for every requirement asked in the project

- 3.1: For Usernames, Password and display names the data type I used is VARCHAR and userNames is my primary key because it is the email id of the person. Also, I have added the UNIQUE constraint to display names so that no two users can have same displayName.

- 3.2: For tweets I am storing the content of the tweet as tweetData whose maximum length is set to 140. tweetTime attribute stores the time stamp of the tweet.

- 3.3: For User relationship management I have created a table called Follows which has two attributes Followin and Follower (explained above) which reference the primary key of User table which is userName.

- 3.4: For User-Tweet relationship management I have used 5 tables (i.e one for each 5 entities connecting Users to Tweet). All the tables have Referencing attributes to userName and Tweet besides their own attributes and primary keys.

- Retweet and CommentsOn tables(relations) basically stores the timestamp and content of retweets and comment respectively.

- Deletes, Likes and Creates all store the attributes referencing primary keys from both the entity sets.

Advantages of my Design

1. It’s simple and easy to understand.
2. No weak entities in the design.
3. All entity sets used in the ER diagram are Unique (no two entity sets have similar entities).

Disadvantages of my design

1. It’s a very basic design and haven’t used many fancy datatypes and constraints which would have made the database efficient.
2. Could have used Serial datatype for assigning IDs for tweetID instead I used bigint which can cause the wastage of space in case of less users and in other case can possibly be not sufficient if no. of users is huge.