Database Project: MessagingApp_DB

Entities

1. **Users**
   The user entity represents the person using the application. It will be described by an id, a name, a phone, an email and a password.

2. **Messages**
   The message entity represents the messages in the application. They will have an a id, a text and the date created which would include the time. It will also contain a number of likes, number of dislikes and number of replies for statistical purposes.

3. **GroupChats**
   The group chat entity represents a group chat in the application. It will contain an id and a group name selected by the owner.

4. **Hashtags**
   The hashtag entity represents a hashtag in a message and it will be used for statistical purposes. It will have an id and a string representing the hashtag.

Relationships

1. *Users* → **Owns** → *GroupChats*
   A user owns a group chat when it is the creator of the group chat. The relationship is **one to many** since a group chat can only have one creator. The group chat have total participation in the relationship since a group chat can't exist without an owner.

2. *Users* → **isMemberOf** → *GroupChats*
   A user can be part of a group chat. The relationship is **many to many** since an user can be in multiple group chats and a group chat can have multiple users.

3. *Users* → **ReactsTo** → *Messages*
   A user can react to a message by liking and disliking a message. The relationship will include an attributes for the **type** of reaction (Like/Dislike). The relationship is **many to one** because a user can only have one reaction to a message, but a message can have a lot of reactions from different users.

4. *Users* → **Owns** → *Messages*
   A user owns a message when it is the author of the message. The relationship is **one to many** since a message can only have one owner. The message have **total participation** in the relationship since a message can't exist without an owner.

5. *GroupChats* → **Has** → *Messages*
   A group chat has messages. The relationship is **one to many** since a group chat can have multiple messages, however, a message can only belong to one group chat.

6. *Messages* → **RepliesTo** → *Messages*
   A message can reply to a message. There are two roles in this relationship, the message reply and the message replied to. The relationship is **many to one** since a message can only reply to one message, but a message can be replied by a lot of messages.

7. *Messages* → **Has** → *Hashtags*
   A message can have a hashtag. The relationship is **one to many** since a message can have multiple hashtags but a hashtag belongs to only one message. The hashtag have **total participation** in the relationship since a hashtag can't exist without a message.