**Description Task**

Answer all questions about Tinder. We are going to draw an ERD and link it to some SQL statements that are straight from our app.

Description scope:

Tinder is a dating application that people can sign up for. The process consists of 3 steps. First you get a screen where you must enter general information about yourself. Your name, your address, your email address, your date of birth, and your gender. You also must write a text about yourself, but we want to keep this short: a maximum of 500 characters. You will then see a screen where you can upload up to a maximum of 8 photos of yourself. Finally, you get a screen with a list of 100 general interests that have been predefined. Here you must indicate 10 that you can find yourself in.

When you open the application, you get a list of people who also use the application. You will then see the information entered for each person (with the exception of the address, email address, and last name.)

You can influence this list of people with 3 settings. You can set a minimum and maximum age, you can indicate which gender you are interested in and finally, you can set a radius in kilometers of how far that other person can be from you. We can retrieve this by addressing the GPS sensor of your device.

Once you have adjusted the settings to your liking, you can go through the list. You have to give each person a 'rating'. This can be: a like (I'm interested), no like (not interested) or a 'superlike' (I'm interested, and the other person can be informed about it.)

If 2 people have given each other a like, there will be a match! Both persons will be notified.

If a match arises between 2 people, these 2 people can send messages to each other in our app. These can be text messages or images. If your message is an image, we would like to include the height, width, and orientation of the image. This is useful to display the images nicely in our chat area.

Finally: a match can also be 'unmatched'. All beautiful songs come to an end.

The customer is very interested in timings. When in the day are the users most active for example? How many matches were 'unmatched' afterwards? Be sure to let us store enough information so that we can provide the customer with analytics from the application.

**Question 1**

Draw the entire ERD. Including entities, attributes (with type!) and connections (with cardinality!).

* Upload file:

**Question 2**

Write the SQL statement that belongs in this function (use the parameters!)

addUser(firstname: String, lastname: String, email: String, brith: Date, sex: String, description: String) {}

**Question 3**

Write the SQL statement that belongs in this function (use the parameters!)

addPicturesToUser(userid: int, pictures: Array of imagepaths) {}

**Question 4**

Write the SQL statement that belongs in this function (use the parameters!)

addInterestsToUser(userid: int, interests: Array of interestids) {}

**Question 5**

Write the SQL statement that belongs in this function (use the parameters!)

You can easily ignore the radius in this question.

getUsers(userid: int, minage: int, maxage: int, sex: String) {}

**Question 6**

Write the SQL statement that belongs in this function (use the parameters!)

sendLike(senderuserid: int, receiveruserid: int) {}