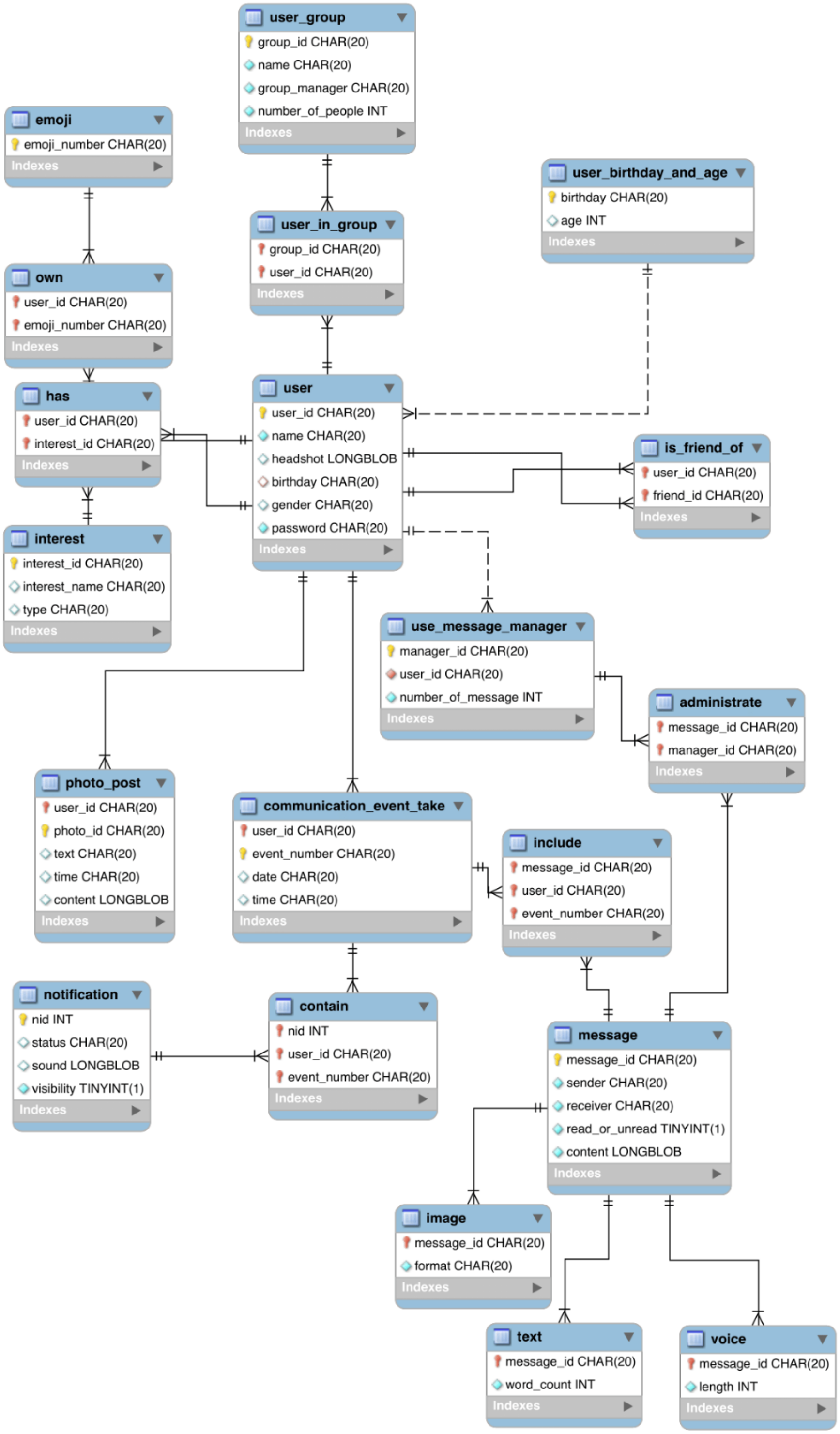
CMPT 354

Chat Application — *LetsTalk*
Database Management System
Milestone 4

Instructor
Hazra Imran

Students:
Ling-Yu Keng #301451848
Sibei Zhou #301416917
Xingjin Li #301386765


user_group
group_id CHAR(20)
name CHAR(20)
group_manager CHAR(20)
number_of_people INT
Indexes
emoji
emoji_number CHAR(20)
Indexes
user_birthday_and_age
birthday CHAR(20)
age INT
Indexes
user_in_group
group_id CHAR(20)
user_id CHAR(20)
Indexes
own
user_id CHAR(20)
emoji_number CHAR(20)
Indexes
has
user_id CHAR(20)
interest_id CHAR(20)
Indexes
user
user_id CHAR(20)
name CHAR(20)
headshot LONGBLOB
birthday CHAR(20)
gender CHAR(20)
password CHAR(20)
Indexes
is_friend_of
user_id CHAR(20)
friend_id CHAR(20)
Indexes
interest
interest_id CHAR(20)
interest_name CHAR(20)
type CHAR(20)
Indexes
use_message_manager
manager_id CHAR(20)
user_id CHAR(20)
number_of_message INT
Indexes
administrate
message_id CHAR(20)
manager_id CHAR(20)
Indexes
photo_post
user_id CHAR(20)
photo_id CHAR(20)
text CHAR(20)
time CHAR(20)
content LONGBLOB
Indexes
communication_event_take
user_id CHAR(20)
event_number CHAR(20)
date CHAR(20)
time CHAR(20)
Indexes
include
message_id CHAR(20)
user_id CHAR(20)
event_number CHAR(20)
Indexes
notification
nid INT
status CHAR(20)
sound LONGBLOB
visibility TINYINT(1)
Indexes
contain
nid INT
user_id CHAR(20)
event_number CHAR(20)
Indexes
message
message_id CHAR(20)
sender CHAR(20)
receiver CHAR(20)
read_or_unread TINYINT(1)
content LONGBLOB
Indexes
image
message_id CHAR(20)
format CHAR(20)
Indexes
text
message_id CHAR(20)
word_count INT
Indexes
voice
message_id CHAR(20)
length INT
Indexes

Project Repository

https://github.com/graykeng/LetsTalk

Project description

We are committed to creating a chat platform called ***LetsTalk*** that allows users to communicate with each other by sending messages. Users need to fill in their profile when creating an account, they can choose their favourite picture as their avatar, and after setting their profile, they will get their UserID. This software is used by UserID and password to log in. After creating your account, you can start chatting. Users can add friends by looking up other users' IDs or become friends with others by joining groups. Messages are divided into two types, text messages and picture messages. Users can use different types of messages to chat. In addition, we also provide an emoji library. In addition to images, users can select emojis from the emoji library directly from the small button next to the chat box. We also provide a picture wall feature where users can add their favourite pictures to the picture wall to browse or save them. If a user is tired of a friend, they can delete it by selecting them in the chat list and then clicking delete on their profile page.

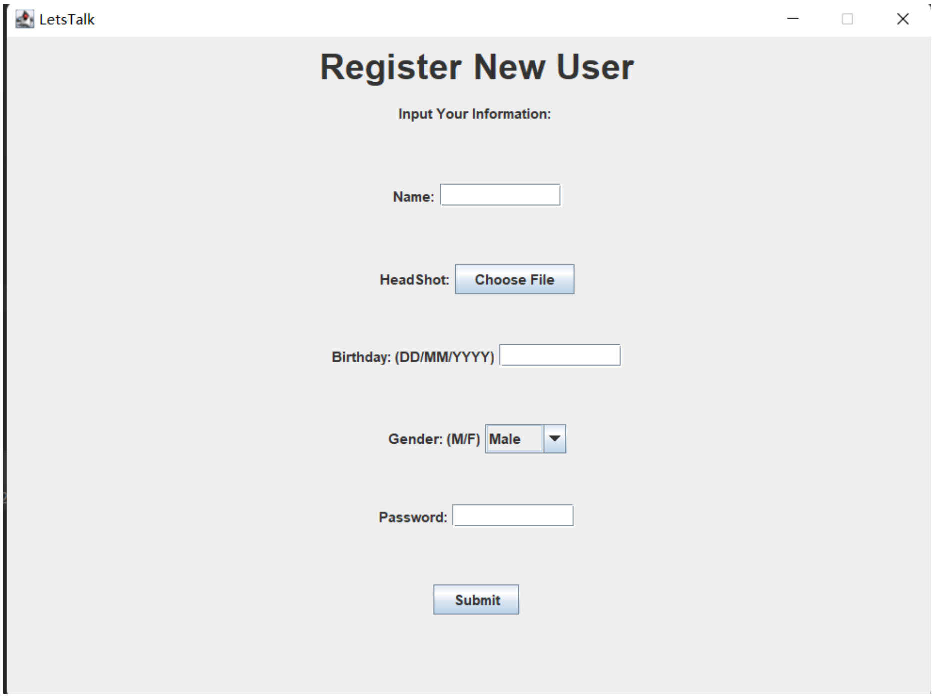
Difference

We removed the feature for the massage manager because it didn't seem to do much. We dismissed the voice messages because it was challenging to implement voice recognition, and we did not have much connection to the database. We also removed the notification feature due to a lack of time.

Screenshot for queries

Insert operation:

Before Insert new user(need to be register):



Register New User

Input Your Information:

Name:

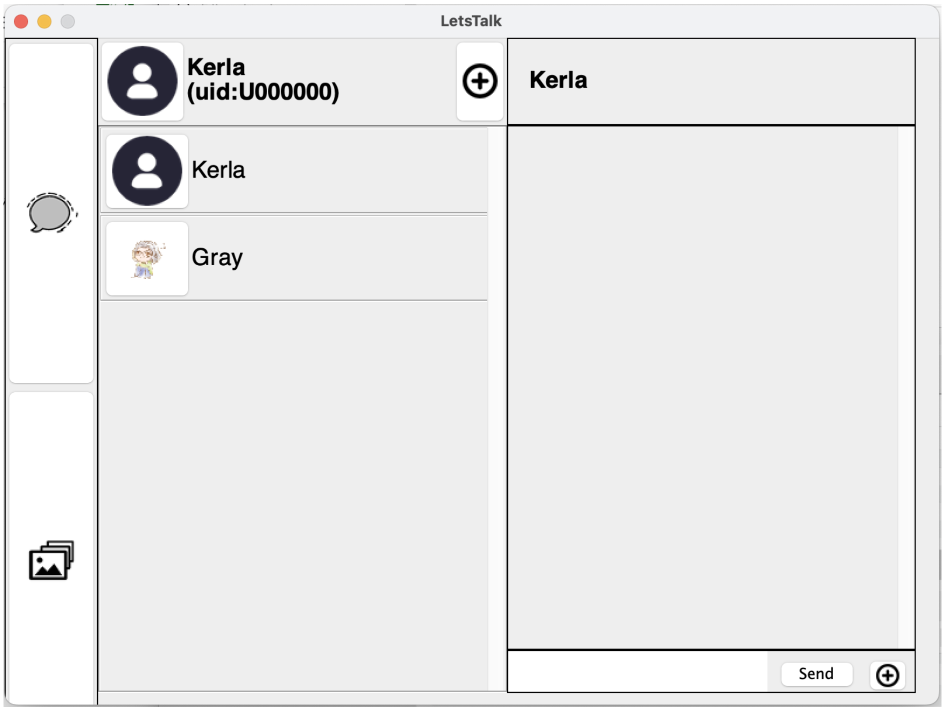
Birthday: (DD/MM/YYYY)

Gender: (M/F) Male

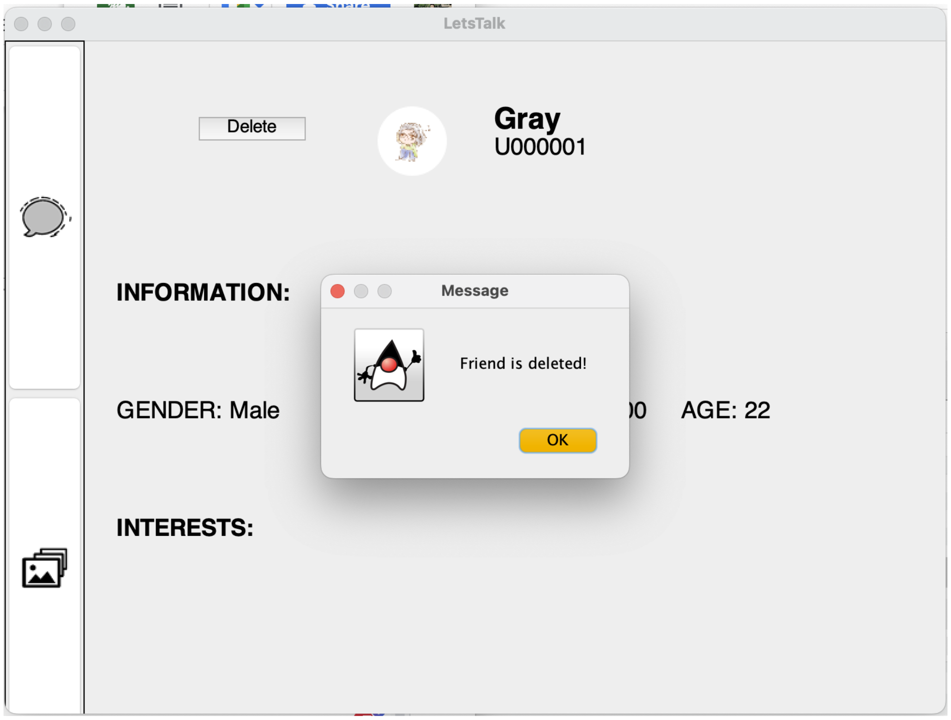
Password:

Delete Operation:

Before the delete is_friend_of operation (user will be a friend of themself, and now only users U000000 and U000001 are friends), after pressing the delete button, the relationship is gone.



Delete operation :



Gray
U000001

INFORMATION:

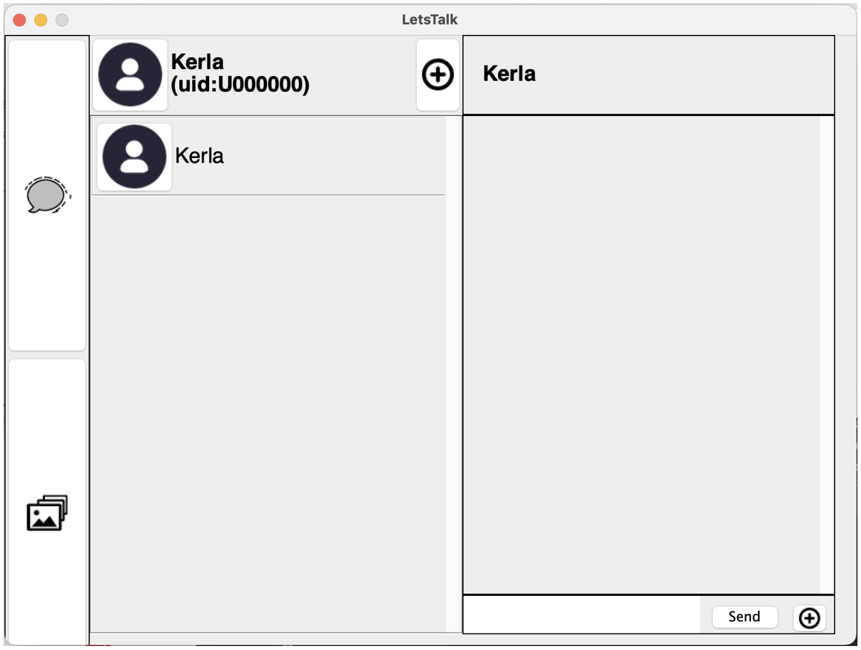
GENDER: Male

00 AGE: 22

INTERESTS:

Friend is deleted!

After the delete operation:



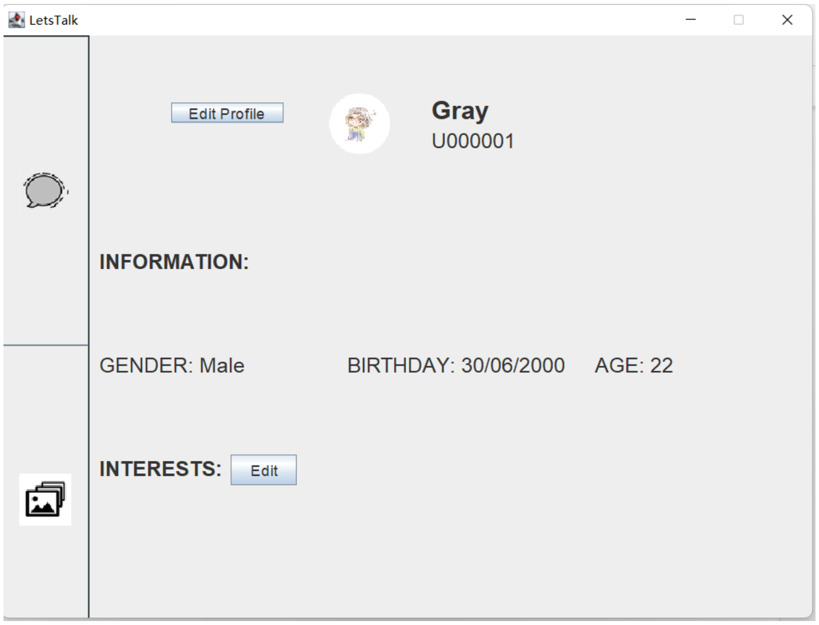
Kerla
(uid:U000000)

Kerla

Kerla

Update operation:

We want to update user_proflie:



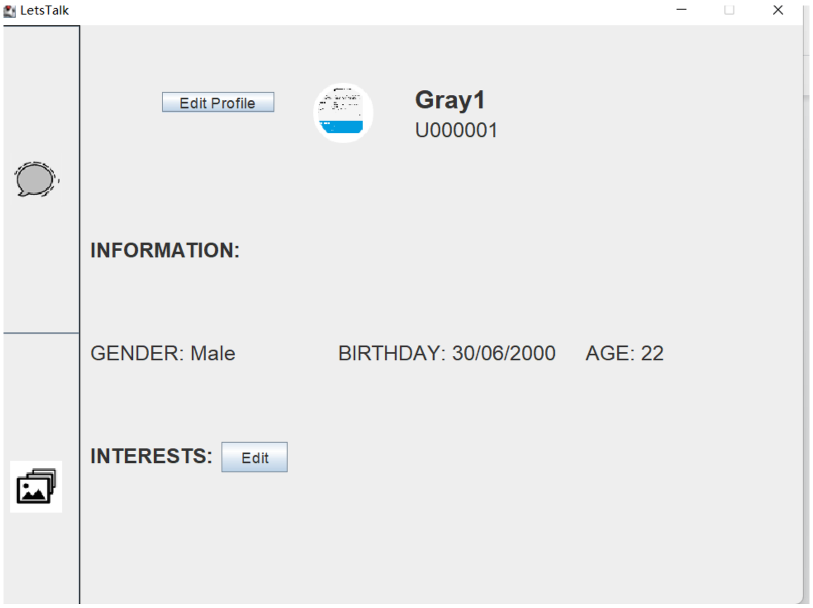
Gray
U000001

INFORMATION:

GENDER: Male BIRTHDAY: 30/06/2000 AGE: 22

INTERESTS:

After pressing edit profile button and resubmit:



Gray1
U000001

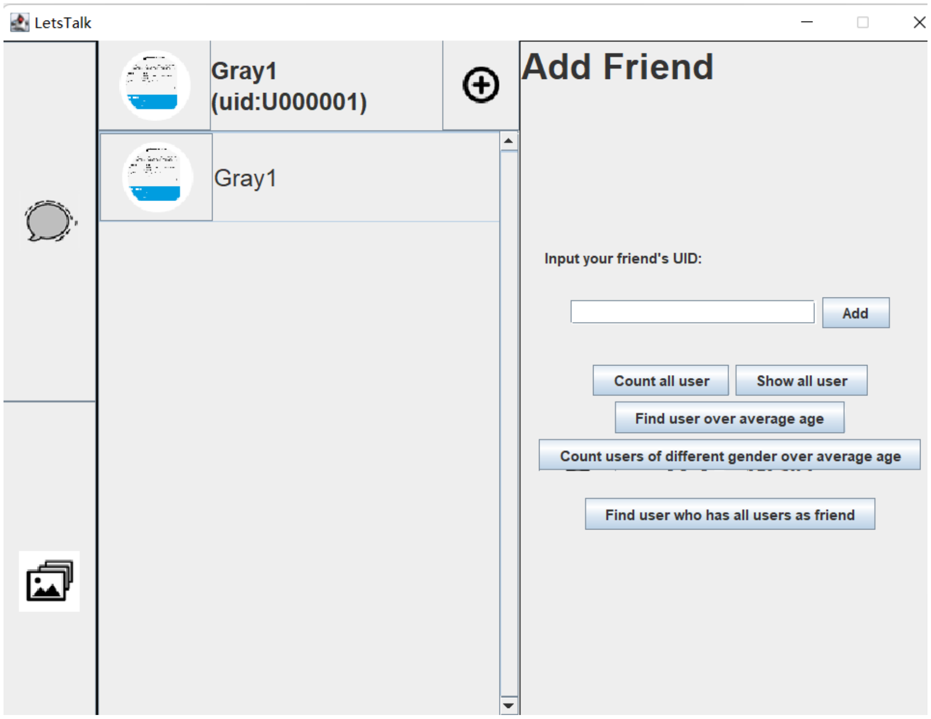
INFORMATION:

GENDER: Male BIRTHDAY: 30/06/2000 AGE: 22

INTERESTS:

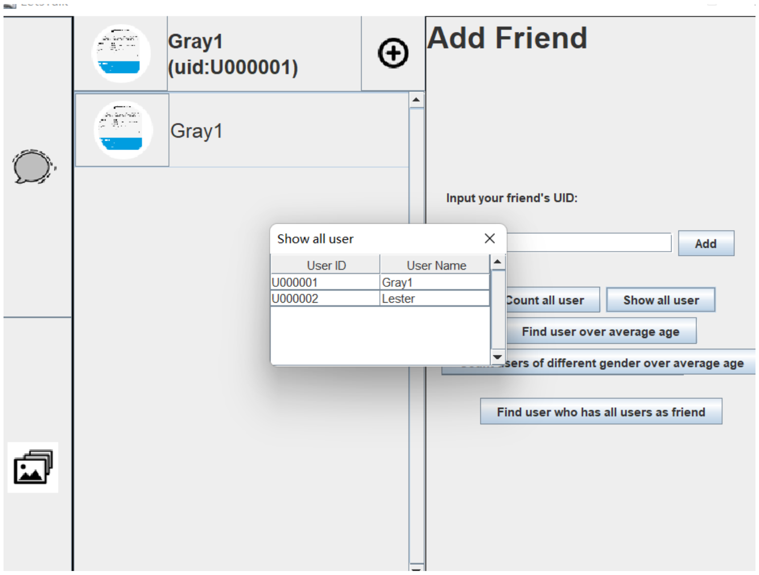
Selection:

Select user_id and add friends with him.


LetsTalk
Gray1
(uid:U000001)
Gray1
Add Friend
Input your friend's UID:
Add
Count all user
Show all user
Find user over average age
Count users of different gender over average age
Find user who has all users as friend

Projection

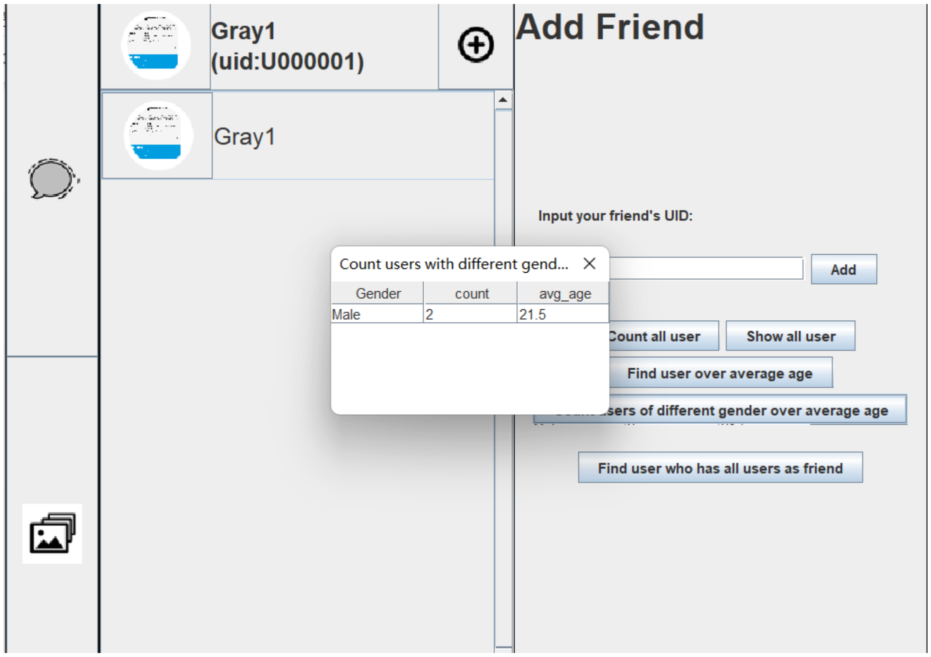
Show all user_id and user_name


Gray1
(uid:U000001)
Gray1
Add Friend
Input your friend's UID:
Add
Count all user
Show all user
Find user over average age
...sers of different gender over average age
Find user who has all users as friend
Show all user

User ID	User Name
U000001	Gray1
U000002	Lester

Join Query :

Find the number of users of different gender over the average age.

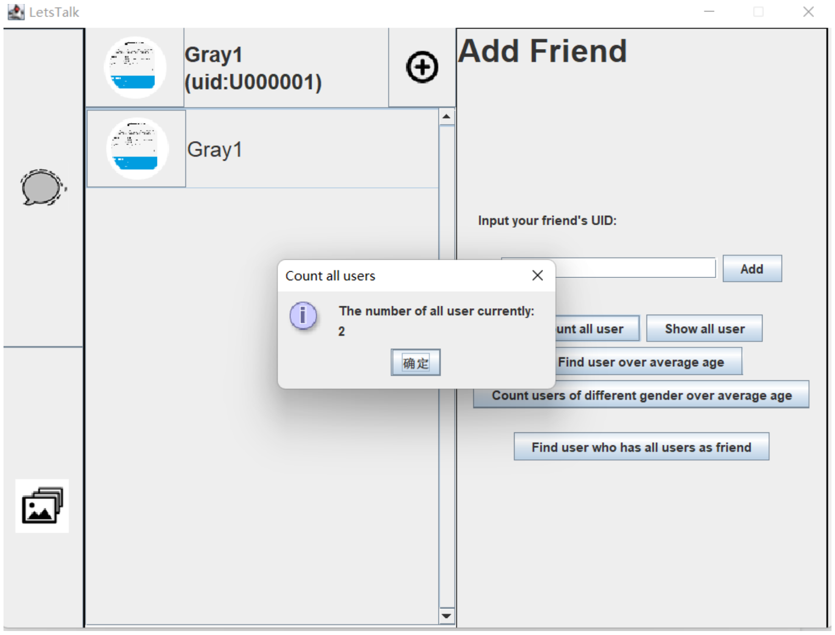

Gray1
(uid:U000001)
Add Friend
Gray1
Input your friend's UID:
Add
Count all user
Show all user
Find user over average age
...sers of different gender over average age
Find user who has all users as friend

Count users with different gend...

Gender	count	avg_age
Male	2	21.5

Aggregation query:

Count the number of users

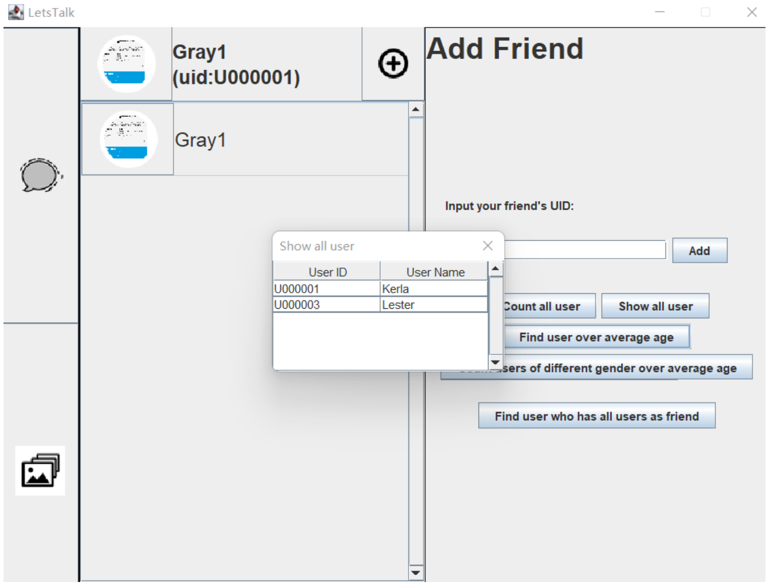

LetsTalk
Gray1
(uid:U000001)
Add Friend
Gray1
Input your friend's UID:
Add
...unt all user
Show all user
Find user over average age
Count users of different gender over average age
Find user who has all users as friend
确定

Count all users

The number of all user currently:
2

Nested Aggregation with group-by:

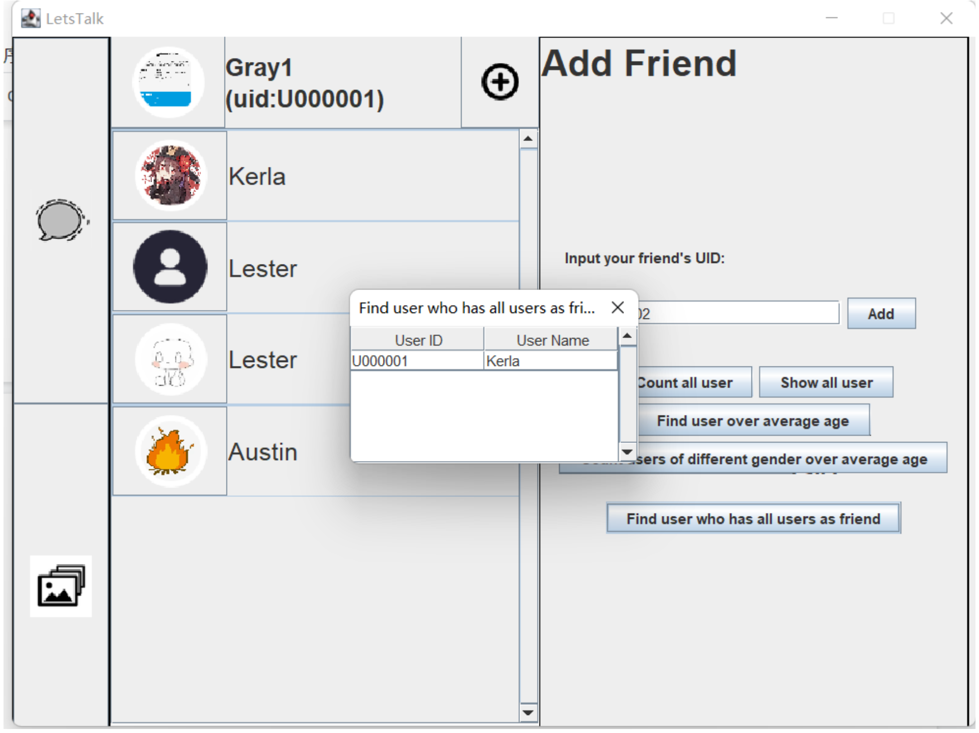
Find users whose age is greater than the average age.



User ID	User Name
U000001	Kerla
U000003	Lester

Division query:

Find users who have all other users as friends.


LetsTalk
Gray1
(uid:U000001)
Kerla
Lester
Lester
Austin
Add Friend
Input your friend's UID:
)2
Add
Count all user
Show all user
Find user over average age
…sers of different gender over average age
Find user who has all users as friend
Find user who has all users as fri...

User ID	User Name
U000001	Kerla