

**Feature:** Manage Follows

**Name:** Michael Kwan

**Queries:**

Fine everyone the user is not following and is not themselves.

```
query = 'SELECT DISTINCT username FROM person WHERE username != %s and username NOT IN \
          (SELECT username_followed FROM follow WHERE username_follower = %s)'
cursor.execute(query, (follower, follower))
```

The user will follow the person in the follow table but the followstatus is set to 0.

```
ins = 'INSERT INTO follow VALUES(%s, %s, 0)'
cursor.execute(ins, (username_followed, username_follower))
```

Determine if there is an entry in the follow table where someone is trying to follow the user.

```
query = 'SELECT DISTINCT username_follower FROM follow WHERE \
          username_followed = %s and followStatus = 0'
cursor.execute(query, username)
```

Accept the follow by setting the followstatus to 1 in that relationship.

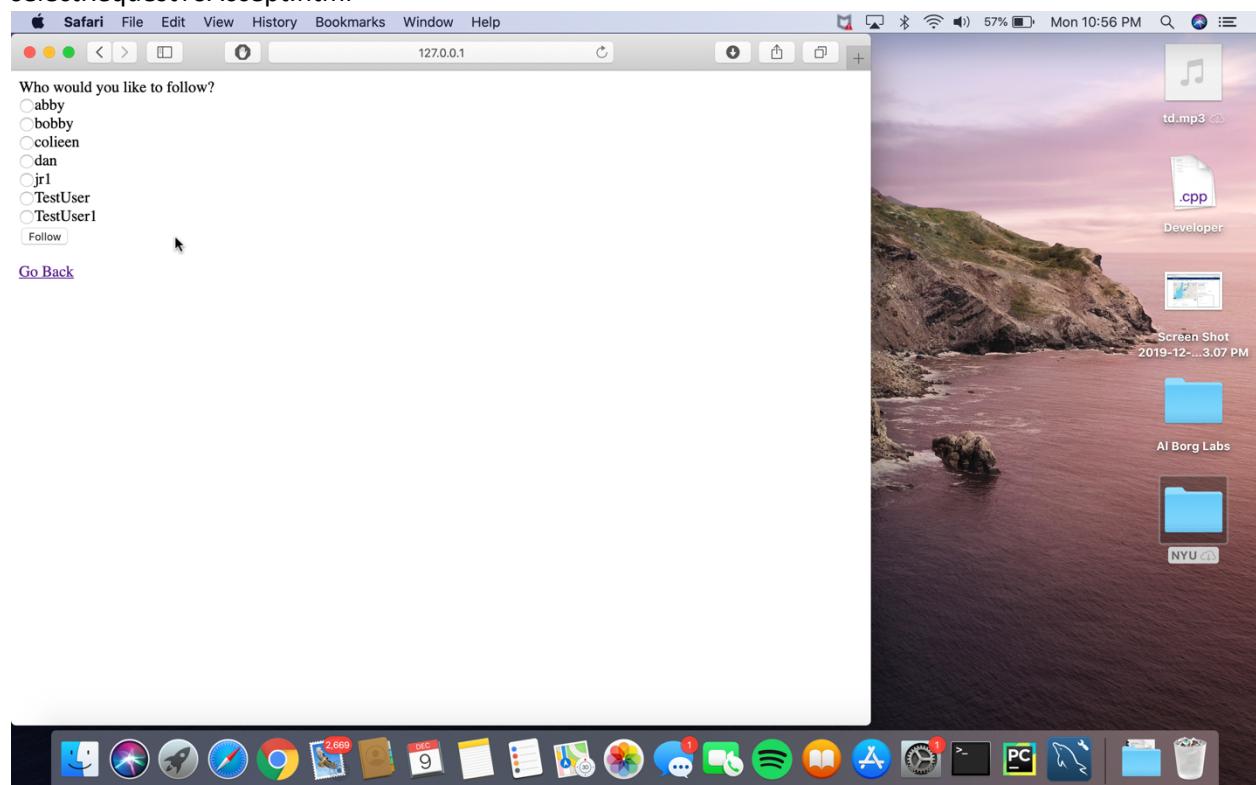
```
upd = 'UPDATE follow SET followStatus = 1 WHERE username_followed = %s AND
username_follower = %s'
```

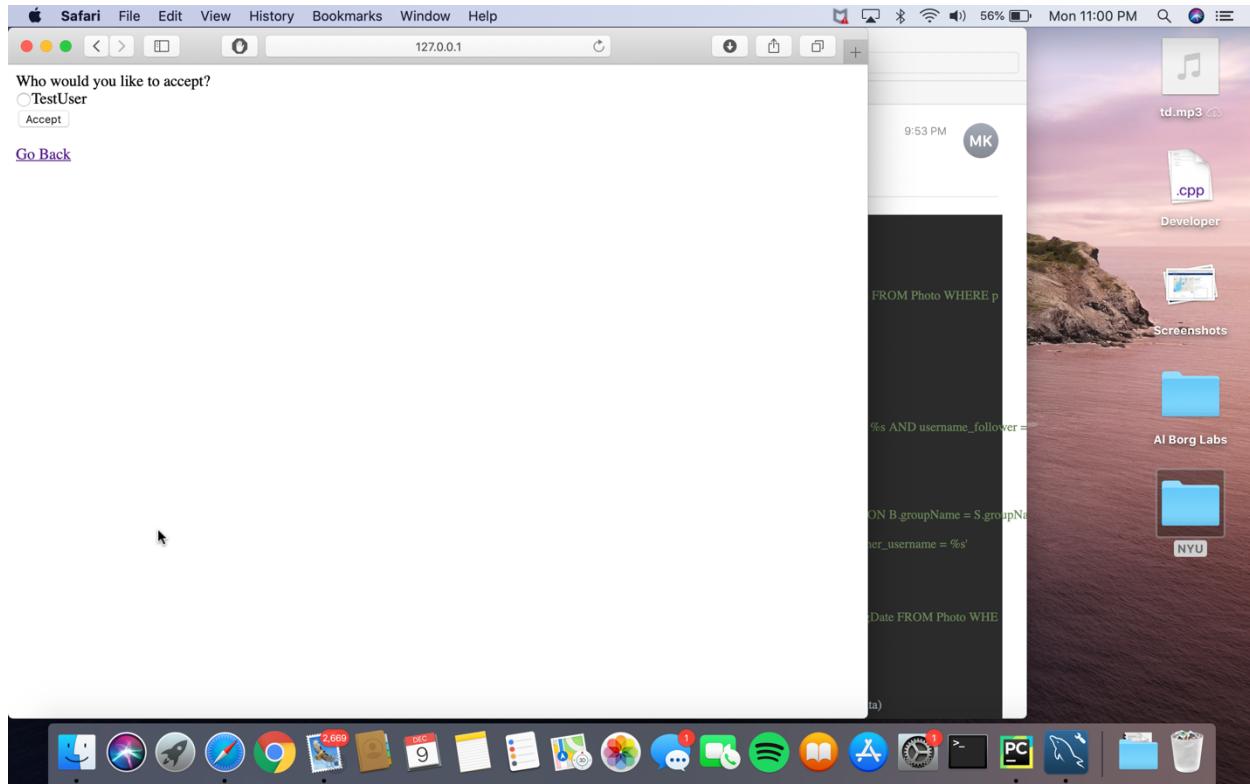
**Locations:**

app.py (Lines 317-372)

select\_Person\_toFollow.html

selectRequestToAccept.html





**Feature:** Manage Tags

**Name:** Nahom Molla

**Queries:**

Accept the tag by changing the tagstatus to 1 in that relationship.

```
upd = 'UPDATE tagged SET tagStatus = 1 WHERE username = %s AND photoID = %s'
cursor.execute(upd, (username, photoID))
```

Determine if there is an entry in the follow table where someone is trying to follow the user and get all the photos the user is tagged in.

```
query = 'SELECT photoPoster, photoID FROM tagged \
        NATURAL JOIN photo WHERE username = %s AND tagStatus = 0'
cursor.execute(query, (user))
```

Determine if the user has already been tagged in the photo.

```
query = 'SELECT * FROM tagged WHERE username = %s AND photoID = %s'
cursor.execute(query, (follower, photoID))
```

Tag the user with tagstatus of 1.

```
query = 'INSERT INTO tagged VALUES (%s, %s, 1)'
cursor.execute(query, (follower, photoID))
```

Determine if the user follows the poster of the photo or is in a friendgroup the photo is shared with.

```
query = 'CREATE OR REPLACE VIEW v AS \
        ((SELECT photoID FROM Follow AS f JOIN \
        Photo AS p on f.username_followed = p.photoPoster \
        WHERE f.username_follower = %s AND f.followStatus = 1) \
        UNION (SELECT photoID \
        FROM (SharedWith AS sw JOIN BelongsTo AS bt ON \
        sw.groupOwner = bt.owner_username AND \
        sw.groupName = bt.groupName) NATURAL JOIN photo \
        WHERE member_username = %s))'
```

```

cursor.execute(query, (follower, follower))
conn.commit()

query = 'SELECT photoID FROM v WHERE photoID = %s'
cursor.execute(query, photoID)
data = cursor.fetchone()
query = 'DROP VIEW v'
cursor.execute(query)

```

Tag the user with tagstatus of 0.

```

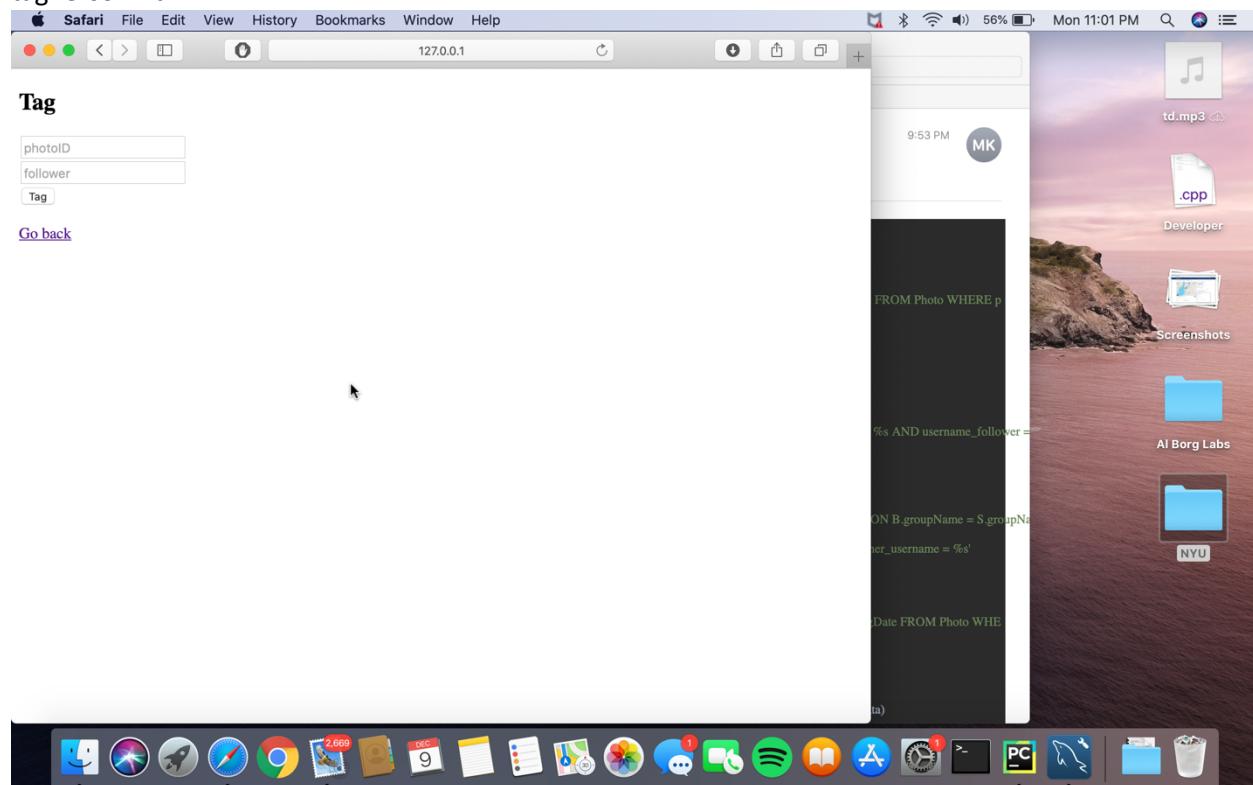
query = 'INSERT INTO tagged VALUES (%s, %s, 0)'
cursor.execute(query, (follower, photoID))

```

### Locations:

app.py (Lines 175-254)

tagPerson.html



**Feature:** Adding friendgroups

**Name:** Michael Kwan

**Queries:**

Determines if a friendgroup with the same groupOwner and groupName already exists.

```

check = 'SELECT * FROM friendgroup WHERE groupOwner = %s and groupName = %s'
cursor.execute(check, (groupOwner, groupName))

```

Inserts the new friendgroup with the groupOwner and groupName into the friendgroup table

Then the groupOwner is added into the that friendgroup as a member in the belongto table

```

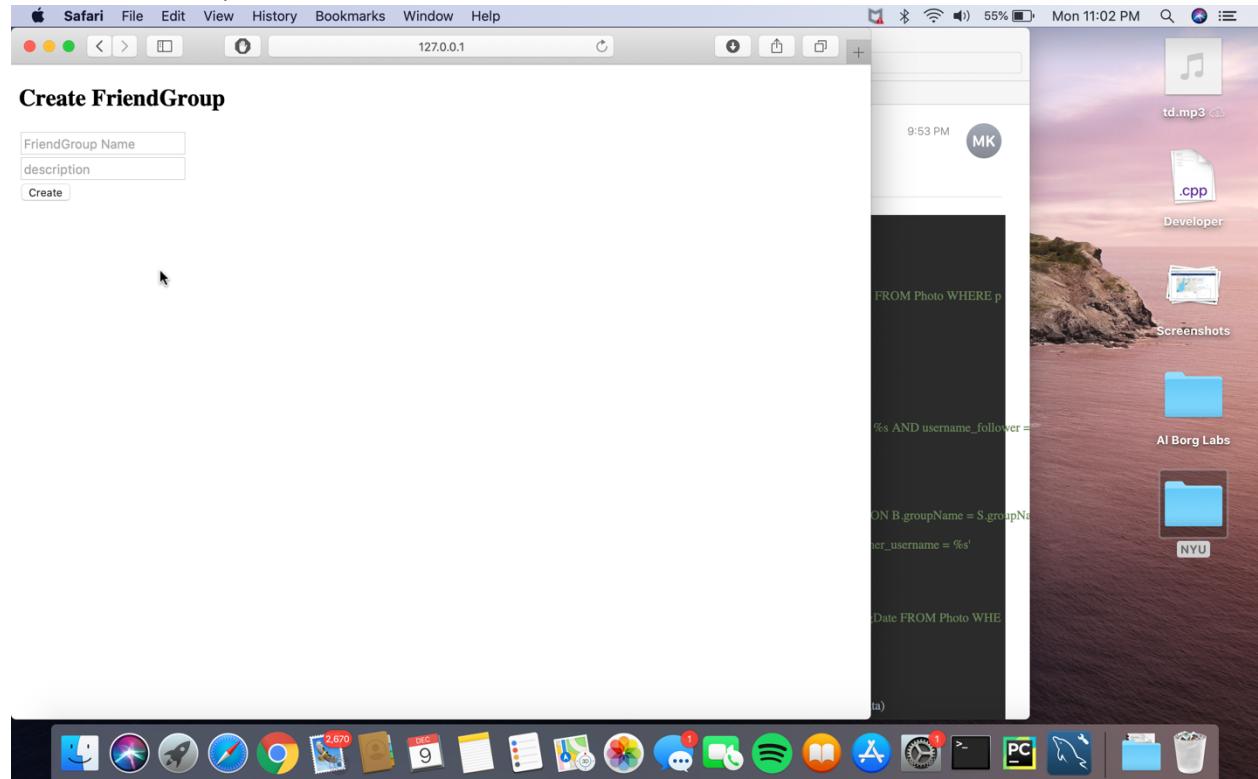
query = 'INSERT INTO friendgroup VALUES(%s, %s, %s)'
cursor.execute(query, (groupOwner, groupName, description))
query2 = 'INSERT INTO belongto VALUES(%s, %s, %s)'
cursor.execute(query2, (groupOwner, groupOwner, groupName))

```

**Locations:**

app.py (Lines 374-399)

createFriendGroup.html



**Feature:** Add friend

**Name:** Michael Kwan

**Queries:**

Check if the friendgroup you want to add a friend to exists in the friendgroup table

```
check = 'SELECT * FROM friendgroup WHERE groupOwner = %s and groupName = %s'  
cursor.execute(check, (owner_username, groupName))
```

Check if the friend username exists in person table

```
check2 = 'SELECT username FROM person WHERE username = %s'  
cursor.execute(check2, (member_username))
```

Check if the friend username is already a member of the friendgroup in the belongto table

```
check3 = 'SELECT * FROM belongto WHERE owner_username = %s and groupName = %s and  
member_username = %s'  
cursor.execute(check3, (owner_username, groupName, member_username))
```

Add the friend into the friendgroup in the belongto table

```
query = 'INSERT INTO belongto VALUES(%s, %s, %s)'  
cursor.execute(query, (member_username, owner_username, groupName))
```

**Locations:**

app.py (Lines 401-439)

addToFriendGroup.html

**Add To FriendGroup**

FriendGroup Name
Username
<input type="button" value="Add"/>

```

9:53 PM MK
FROM Photo WHERE p
%$ AND username_follower =
ON B.groupName = S.groupName
username = %s'
Date FROM Photo WHE
(a)

```

**My Photos**

Username	Photo ID	Caption	Time Posted
nym228	1	ss	2019-11-24 13:48:41

**My FriendGroups**

**People In My Groups:**

Group Name	Member Username
------------	-----------------

**In Groups:**

Owner Username	Group Name
TestUser	FriendGroup1

[Follow](#)  
[See Follow Requests](#)  
[Search Photos By Poster](#)  
[Tag a Person in a Photo](#)  
[View Tags](#)  
[Create FriendGroup](#)  
[Add to FriendGroup](#)

[Logout](#)

**Feature:** Search by poster

**Name:** Nahom Molla

**Queries:**

Determine if the user is the photoPoster and if so, get all the info about that photo.

```
query = 'SELECT photoPoster, photoID, caption, postingDate FROM Photo WHERE  
photoPoster=%s ORDER BY postingDate DESC'  
cursor.execute(query, (user))
```

Determine if the user is following the poster.

```
query = 'SELECT * \  
        FROM follow \  
        WHERE followstatus = 1 AND username_followed = %s AND username_follower = %s'  
cursor.execute(query, (poster, user))
```

Get all the photos that the photoposter posted and has allFollowers = 1.

```
query = 'SELECT photoPoster, photoID, caption, postingDate FROM Photo WHERE  
photoPoster=%s AND \  
        allFollowers=1 ORDER BY postingDate DESC'
```

```
cursor.execute(query, (poster))
```

Determine if the user is in the friendgroup the photos are shared with.

```
query = 'SELECT * \  
        FROM belongto AS B JOIN sharedwith AS S ON B.groupName =  
S.groupName AND B.owner_username = S.groupowner\  
        WHERE member_username = %s AND B.owner_username = %s'  
cursor = conn.cursor()  
cursor.execute(query, (user, poster))
```

Get all the photos that the photoposter posted and has allFollowers = 0.

```
query = 'SELECT photoPoster, photoID, caption, postingDate FROM Photo WHERE  
photoPoster=%s \  
        AND allFollowers = 0 ORDER BY postingDate DESC'  
cursor.execute(query, (poster))
```

Get all the photos that the photoposter posted.

```
query = 'SELECT photoPoster, photoID, caption, postingDate FROM Photo WHERE  
photoPoster=%s \  
        ORDER BY postingDate DESC'  
cursor.execute(query, (poster))
```

#### Locations:

app.py (Lines 256-314)

searchByPoster.html

Safari File Edit View History Bookmarks Window Help

127.0.0.1

Search poster

Username Photo ID Caption Time Posted

Username	Photo ID	Caption	Time Posted
nym228	1	ss	2019-11-24 13:48:41

[Go back](#)

9:53 PM MK

```
FROM Photo WHERE p
```

%s AND username\_follower =

ON B.groupName = S.groupName

username = %s'

Date FROM Photo WHE

(a)

Mon 11:01 PM

Safari File Edit View History Bookmarks Window Help

127.0.0.1

Search poster

Username Photo ID Caption Time Posted

Username	Photo ID	Caption	Time Posted
nym228	1	ss	2019-11-24 13:48:41

[Go back](#)

9:53 PM MK

```
FROM Photo WHERE p
```

%s AND username\_follower =

ON B.groupName = S.groupName

username = %s'

Date FROM Photo WHE

(a)

Mon 11:01 PM

Safari File Edit View History Bookmarks Window Help

127.0.0.1

Search poster

Username Photo ID Caption Time Posted

Username	Photo ID	Caption	Time Posted
nym228	1	ss	2019-11-24 13:48:41

[Go back](#)

9:53 PM MK

```
FROM Photo WHERE p
```

%s AND username\_follower =

ON B.groupName = S.groupName

username = %s'

Date FROM Photo WHE

(a)

Mon 11:01 PM

Safari File Edit View History Bookmarks Window Help

127.0.0.1

Search poster

Username Photo ID Caption Time Posted

Username	Photo ID	Caption	Time Posted
nym228	1	ss	2019-11-24 13:48:41

[Go back](#)

9:53 PM MK

```
FROM Photo WHERE p
```

%s AND username\_follower =

ON B.groupName = S.groupName

username = %s'

Date FROM Photo WHE

(a)

Mon 11:01 PM