

Overview:

For this project, we used in combination several data files (.csv) and sql files (.sql) to implement our database and populate it with data.

We have 4 distinct .sql files

1. create_triggers.sql
 - a. used to create procedures/functions to implement the required functionality.
 - b. implement server side processing
 - c. implemented the following functions
 - i. newAccount(userid,password,email);
 - ii. login(userid, password);
 - iii. newconnection(userid, victimid);
 - iv. status
 - v. acceptConnection
 - vi. rejectConnection
 - vii. getFriends
 - viii. getWork
 - ix. getEdu
 - x. SearchUser
 - xi. userData
2. load_data.sql
 - a. used in combination with the data files to load data into the database.
3. create_tables.sql
 - a. used to create the tables along with all the required constraints.
4. create_index.sql.
 - a. used to create indices to improve the performance of the queries that were implemented.

We use these functions within our ProfNetwork.java file to to do our queries(insert, count, delete, etc). In addition, we have the following .csv files

1. Connection.csv
2. USR.csv
3. Edu_det.csv
4. Message.csv
5. Work_ex.csv

These files contain all of the data we used for testing and our initial database implementation.

We use a compile.sh script to run the database. Once the script is run, a terminal menu appears that allows the user to communicate with the db.

Task distribution:

Robert Pasillas:

```
create csv(conection, usr, edu_det, message, work_ex)
created tables(connection, message, educational_deatils)
load_data(connection, message, educational_details)
created index(usr_userid, con_user, con_con, msg_sndr, msg_rcvr, wrk_usr, edu_usr)
```

Selik Samai:

```
created tables (usr, workexpr)
load_data(usr, workexpr)
```

ProfNetwork.java

These are the functions we implemented and who worked on what

1. CreateUser(esql) [Robert]
2. LogIn(esql) [Robert]
3. ChangePassword(esql)[Robert]
4. deleteAccount(esql) [Selik]
5. FriendList(esql, authorisedUser) [Robert]
6. UpdateProfile(esql) [Robert]
7. NewMessage(esql) [Selik]
8. SendRequest(esql) [Selik]
9. OtherFriendList(esql) [Selik]
10. Profile(esql, authorisedUser) [Robert]
11. otherProfile(esql) [Robert]
12. Summary (esql) [Selik]
13. Read(esql) [Selik]
14. Sending(esql) [Selik]
15. delete(esql) [Selik]

Implementation of triggers and many sql functions were done by Robert.