Test Data and Test Document

Class: CS425 - Database Design and Applications

Project: Interactive Students/Faculties Networking

Team member: Danna Liu (A20345502)
Shanshan Jiang (A20354599)
Zhizheng Li (A20351352)

Created by Shanshan Jiang
Revised by Danna Liu

Introduction:

- **Test document**
  For each requirement in the "General Processes" section, the testing document will include all the following lists:
  _ List the requirement
  _ List all tests necessary to perform on that requirement
  _ List the expected outcome
  _ List the actual outcome

  _ Indicate if the implemented requirement passed all testing

- **Test data**
  The test data will also be provided for the above test.

General Processes:

1. Demonstrate the process of register into an interest group.

This requirement asks us to register into an interest group. We have to perform the following test to satisfy it.

- The user **jeff2016** who wants to register into an interest group must be a registered student/faculty. If yes, he/she could start to register into an interest group.
  If not, he/she must first to register as a student/faculty.
- There is at least one existed group with assigned moderator for the user to register into.
- The user must be known which group **cs425_spring2016** he/she are going to sign up and input their group name **jeff425**, password **1234** and group name **cs425_spring2016** to register.
- After registration for an interest group, the user must be approved by the moderators **david425 or dora425** of that group before logging into the group.
- Once approved, the user **jeff425** could log into the group.

.

Expected Outcome:

- An unregistered student/faculty **ann2016** would be rejected to sign up a group. A registered student/faculty **jeff2016** successfully signs into an interest group after approving by moderator **david425 or dora425** of that group.
- He/she could user his/her user name and password to login into the group registered.
- The data related to a new group user is stored into table ***groupUsers***.

Actual outcome:

Our code allow us to achieve all above expected outcome.

2. Demonstrate the process of posting comments and discussions. The process will check that the posting will fail if the user is not registered to the course or group.

This requirement asks a group user to posting comments.

- The user who want to post a comment must be a group user for that group. That is, the user **jeff425** have to log in a group first and then post comments.
- There must be at least one existed forums or open a new forum for him/her to post comments. Suppose **jeff425** post on an existed forum **forum-1 for cs425** to post comments **"hello, I am Jeff."**

Expected Outcome:

- Both an unregistered user **ann2016** and a group user **mark450** from group **cs450_spring2016** are not allowed to sign in the group **cs425_spring2016.** Only the group user **jeff425** is able to sign in a group, and see all other comments in that group.
- Post the comments **"hello, I am Jeff."** successfully.
- The data related to the comment he/she posted is stored into table ***Comments.***

Actual outcome:

Our code allow us to achieve all above expected outcome.

3. Demonstrate the process of assigning a TA to a course and the assignment will fail if the person is also a student of the course.

This requirement asks admin to assign a course TA who must not be a student of that course.

- There must be at least one existed course for the assignment.
- The **admin** must be first log into the system and then try to assign a student **jeff2016** who is not a student of the course **cs450** as a TA.
- Try to assign a student who is a student **tina2016** as a TA of that course to see if the assignment will fail.

Expected Outcome:

- A student **jeff2016** who is not a student of that course could be assign as TA of that course **cs450**.
- A student who is a student **tina2016** of that course **cs450** could not be assign as TA of that course.
- The data related to the TA is stored into table ***TAs***.

Actual outcome:

Our code allow us to achieve all above expected outcome.

4. Demonstrate the process of opening a discussion forum for an interest group.

This requirement asks a group user to open a discussion forum for an interest group.

- There must be at least one existed group.
- The user **jeff425** who wants to open a discussion forum **forum-8 for cs425** for an interest group **cs425_spring2016** must be a group user for that group. If not, he/she is not able to log into a group. As result, there is no possible for him/her to open a discussion forum.
- Log into a group **cs425_spring2016** first, and then click "add forum" to open a forum.

- Now, this forum is not approved, the group user should wait for the moderator **david425 or dora425** to approve it and give a bonus **5** to him/her.

Expected Outcome:

- A group user **jeff425** can open a discussion forum **forum-8 for cs425** successfully and earn a bonus **5** from this performance.
- The data related to the opening of a discussion forum is stored into table ***discussionForums***.

Actual outcome:

Our code allow us to achieve all above expected outcome.

5. Demonstrate the process of modifying and filtering messages.

This requirement asks admin or moderators to modify and filter messages.

- The **admin** or moderator must be first log into the **system or groups** and then try to modify and filter messages such as add/delete/update all contents of users.
- The moderators **david425 or dora425** must be first log into the group **cs425_spring2016** and then try to modify and filter messages such as add/delete/update all comments or discussion forums such as **deleting forum-8 for cs425)**.

Expected Outcome:

- Both **admin and moderators david425 or dora425** are able to finish the process of modifying and filtering messages **such as delete forum-8 for cs425**.
- All modified content would be stored into the database. For example, **there is no forum-8 for cs425 in the database.**

Actual outcome:

Our code allow us to achieve all above expected outcome.