Project Outline

Database Systems
Computer Science 370
Summer 2016
University of Victoria

Project Due Dates:

Monday, May 16:	**Team Submission (5% of project mark)**
Thursday, June 9:	**E/R Diagram and Progress Report (45%)**
Monday, July 18:	**Project Final Report (15%)**
TBD:	**Project Demos of Final Product (35%)**

NOTE: **Project to be done in groups of 2-3 people**
All submissions must be in PDF format

In this project you will design a database for the following application, implement the design in MySQL, and test your implementation with data provided by yourself and (possibly) with a provided data dump.

The application is called "Saiddit". It allows accounts to post text and links to a given "subsaiddit" (or sub community) and allows other accounts to up-vote, down-vote, and comment on the posts, etc. Your requirements elicitation team has extracted the following information:

- An account has a username, password (salted hash), and total reputation (upvotes - downvotes).
    - An account can make posts to any subsaiddit.
    - An account can make comments on a post.
    - An account can have friends (other accounts).
    - An account can have favourite posts.
    - An account can subscribe to a subsaiddit.
- A post has a date/time published, a date/time edited (if it has been), a title, a url (if it is a link post), the post's text, number of upvotes, number of downvotes, and what subsaiddit it belongs to.

- A comment has a date/time created, the text of the comment, the post it refers to, number of upvotes, number of downvotes, if it is a reply the post's parent, and the creator of the comment.
- A subsaiddit has a title, description, who and when they created it, and it may or may not be a default subsaiddit (default meaning that it will show on the "front-page" when not logged in).
- An account can either upvote or downvote but not both and only once per post/comment.

Project Deliverables

- **Team Submission (Due Monday, May 16, 11:59pm) (TOTAL 5%):**
  - Read the project outline
  - Answer the following questions:
    - How many people are on your team (MAX 3)?
    - What are their names, student numbers, and e-mail addresses?
    - Summarize the project outline in 5 sentences.
- **E/R Diagram and Progress Report (Due Thursday, June 9, 11:59pm) (TOTAL 45%)**
  - Provide an E/R diagram for your project solution (35%)
  - Progress Report (10%)
    - Will expect something to be done
    - What has been done?
    - What is left to do?
    - Who did what? (Everyone must do something)
    - What are the issues you ran into?
    - **1 page MAX**
- **Final Report (Due Monday, July 18, 11:59pm) (TOTAL 15%)**
  - Very quick intro
  - How far did you get? If you didn't complete, why?
  - What did you do since progress report?
  - Who did what?
  - What are the issues you ran into? (Extend from progress report but don't copy-paste)
  - Include table-creation statements
  - Include table population statements (Manual ones)
  - **1-2 pages MAX (Minus table-creation and population statements)**
- **Final Product (TOTAL 35%)**
  - Will be based on a demo of the product at a to-be-determined time.

Expected Project Final Product

1. Draw E/R diagram for your solution.
2. Passwords must be stored in a salted hashed format (no plaintext passwords)
3. Translate your E/R diagram into tables in MySQL and include the necessary constraints.
4. Create forms to populate (put data into) your tables.
5. Create a logged-in "front-page" for a given account. A front-page shows the top-voted posts for an account's subscribed subsaiddits.
6. Create a "default" front-page (not logged in) that includes only the top-voted posts from the default subsaiddits.
7. Create a form to delete a post (and all of its comments)
8. Create and execute the following SQL queries (Will be tested in demo):
    a. Get all of the posts by account A, sorted by highest rating (upvotes - downvotes).
    b. Get all of the posts from account A's friends, sorted by highest rating.
    c. Get account A's subscribed subsaiddits (include the default subsaiddits)
    d. Get account A's favourite posts
    e. Get account A's friend's favourite posts
    f. Get account A's friend's subscribed subsaiddits (no duplicates!)
    g. Get all of subsaiddit S's creator's posts
    h. Get all of the posts in subsaiddit S that contain <some text> (Very basic search)
9. What query in 7 took the most time to execute? Why?