TDT4145 Project Spring 2021

The project will be evaluated by the teachers and will count 40 % of your final grade. It should be done in groups of max 3 students.

The learning outcome of the project is the following:

1. Get experience in data modelling and translation into SQL
2. Get practical experience in programming towards a SQL database using JDBC
3. Make concise documentation of high quality

The project is to create a database for storing Piazza type of discussions. Piazza has users which are either Instructors or Students. Those type of users have different permissions. Instructors may manage Students by inviting both professors and students to participate in the forum for a specific course. An instructor may decide if discussions allow anonymous posts, or if posts should be identified publicly by the user's name. An instructor may manage folders and subfolders in Piazza. The folders may be created, edited and removed by instructors. Folders are used to group posts. Each user logs into to the system by an e-mail address and a password.

Each course has a name, a term, multiple users, some folders and a sequence of threads. Posts are organized in threads, where students or instructors may reply to questions, or may make additional comments or follow-up discussions. Posts may be tagged as questions, announcements, homework, homework solutions, lectures notes and general announcements. A thread (i.e., the first post) must be categorized to which folder it belongs, e.g., like exam, exercise 1, etc. Each post may have "likes" (named "good comment"). This may be given by each user. A post does also get a color code depending on if it has a reply or not, and it has a different color code depending on if it is an instructor which has replied, or if it is a student.

The functionality of the system is to make posts or reply to posts and to search for keywords in posts. A user may also insert "good comment" to posts. An important functionality is statistics, which is available to instructors. It shows how many users have been active each day, it also shows which users are most active, by showing the number of posts they have viewed and created. It should also be possible to see a rank/sorting of the most active threads, i.e., the threads being most viewed and commented. It is possible to create links between posts, such that a previous reply may be referenced when the same question or a similar question is posed.

The following use cases should be designed and implemented in the project. It is ok if all use cases are within a fixed course.

1. A student logs into the system, i.e., check user name and password (you do not need to encrypt/decrypt passwords). This should have e-mail and password as input, and these should match this info in the database.
2. A student makes a post belonging to the folder "Exam" and tagged with "Question". Input to the use case should be a post and the texts "Exam" and "Question".
3. An instructor replies to a post belonging to the folder "Exam". The input to this is the id of the post replied to. This could be the post created in use case 2.
4. A student searches for posts with a specific keyword "WAL". The return value of this should be a list of ids of posts matching the keyword.
5. An instructor views statistics for users and how many post they have read and how many they have created. These should be sorted on highest read posting numbers. The output is "user name, number of posts read, number of posts created". You don't need to order by

posts created, but the number should be displayed. The result should also include users which have not read or created posts.

If you are in doubt about anything in this description, just log into Piazza and check for functionality. You don't have to include functionality for including images in Posts, text is enough. It is also enough to have a fixed limit of how long a text may be. E.g., use VARCHAR(500) or similar. You don't have to have any fancy text features, just plain text. For user names and passwords, it is not necessary to use any encryption in this project. Just store email addresses and passwords as text.

If you make assumption beyond the problem text, you should explicitly state which assumptions you make.

There are two deliverables in this project:

DB1

Delivery 1: Conceptual data model

Deadline March 12th:

a) An ER-model showing your complete data model.

b) The ER-model translated to relational schemas (tables). The relational schema should be on 4th normal form. You should explain why this schema is on this normal form. In case it is not, you should explain why it isn't.

c) A description that explains how your model satisfies the 5 usecases listed above. This means that you should check that all information required in the queries are covered by the tables. For every numbered usecase in the task description, explain how your model can solve the problem and satisfy the desired functionality.

d) An SQL script that constructs the database with tables. Remember primary and foreign keys, as well other restrictions if you find necessary. Save your answers in a PDF-document. The SQL script should be attached as a .txt-file.

Let the document be concise and the figures be easy to understand.

Remember to write the names of all group members in your delivery, both in your PDF-delivery and on BlackBoard.

DB2

Delivery 2: The Piazza database implemented in Java with JDBC.

The database should be implemented in Java based on the schema given in the first delivery. The 5 requirements for desired functionality must be fulfilled. You can either make a graphical user interface or a simple text-based interface. Remember that the goal of the project is to practice SQL and JDBC (Java Database Connectivity), and not the Java programming itself.

Deadline March 25th:

a) A runnable program as a JAR-file.

b) Well-documented source code in a zip-file or similar.

c) A textual description that documents your application, delivered as a PDF. The documentation must contain

a. A list of which classes exist in your program and a corresponding description of what task that class solves.

b. An overview of the use cases that have been solved and how they are realized in your program.

Let the document be concise and the figures be easy to understand.

Criteria for evaluating the project

Here are the criteria which are used to evaluate the project:

DB1

1. The use of entities vs. relationships vs. attributes. “Correct” level of use of these concepts. Right number of entities and relationships.
2. Use of keys. Natural vs. generated keys.
3. Constraints in the model, e.g., cardinalities. Are they used correctly?
4. Mapping to SQL tables. Correct use of SQL. Correct use attribute domains. Use of key constraints and Unique. Use of foreign keys constraints. Understanding of 4NF.
5. A description of how the different use cases maps to the different tables. Which tables are used in the different use cases. Is all needed information contained in the tables.
6. The documents should be concise and the figures should be clear.

DB2

1. An overview on how the use cases are solved. This could be textual, or by use of some diagrams.
2. Correct use of JDBC connected to Java
3. Understandable and readable code
4. Concise documents and clear and easily understandable figures.

With respect to the marks used, we rely on the general description:

General description of valuation criteria, English

Symbol	Description	General, qualitative description of valuation criteria
A	Excellent	An excellent performance, clearly outstanding. The candidate demonstrates excellent judgement and a high degree of independent thinking.
B	Very good	A very good performance. The candidate demonstrates sound judgement and a very good degree of independent thinking.
C	Good	A good performance in most areas. The candidate demonstrates a reasonable degree of judgement and independent thinking in the most important areas.
D	Satisfactory	A satisfactory performance, but with significant shortcomings. The candidate demonstrates a limited degree of judgement and independent thinking.
E	Sufficient	A performance that meets the minimum criteria, but no more. The candidate demonstrates a very limited degree of judgement and independent thinking.
F	Fail	A performance that does not meet the minimum academic criteria. The candidate demonstrates an absence of both judgement and independent thinking.