CS480 – Course project

Summer 2020

Database: Knowledge Management

Description:

Consider the design of the following database system for a knowledge management system: a content with a certain content id, name and decription is created by the author. The content is then reviewed by the reviewer and activated for the users to see. Each content has a number of pages stored in the content pages table. It contains the page number, type(PDF, HTML etc) and the path of the page. Each user reads the content and this activity is stored in the user_activity table. The time of the transaction is also stored.

Project Part 2 – CRUD (Create, read, update, and delete)

Deadline: July 18, 2020

List of entities:

1. Author
2. Reviewer
3. content_pages
4. user
5. content
6. user_activity

Based on the Demo (Part 1), implement the following functionality using Java and SQL with necessary GUI interfaces.

1. Insert/delete/update/read a **Author** (all attributes except the author id). The author id should be generated by the system automatically using MySQL autoincrement.

2. Insert/delete/update/read a **reviewer** (all attributes except the reviewer id). The reviewer id should be generated by the system automatically using MySQL autoincrement.

3. Insert/delete/update/read a **user** (all attributes except the user id). The user id should be generated by the system automatically using MySQL autoincrement

4. Insert/delete/update/read a **content** (all attributes except the content id). The content id should be generated by the system automatically using MySQL autoincrement.

5. Insert/delete/update/read a **content_pages** (all attributes except the page id). The page id should be generated by the system automatically using MySQL autoincrement.

6. Insert/delete/update/read a **user_activity** (all attributes).

Project Part 3 – Queries

Deadline: August 1, 2020

Based on the Demo, implement the following functionality using Java and SQL with necessary GUI interfaces.

Trivial Queries:

1. List all Content
2. List all Authors
3. List all Users

Non-trivial Queries:

1. List all the content pages by a particular content type.

2. List all the users with progress 100 for a content.

3. List all users that have not seen a particular content.

4. List the user with the top content consumption rate. (Alphabetical if there is a tie).

5. Track and monitor the progress on the fly for a user, create a view which contains the progress of all the users by the name ‘progress’.