Caroline Liu

Achraf Mamdouh

Yahya Yacoubi

CSC 3326: Implementation Progress

April 9, 2017

The Show Me AUI database keeps track of users, pictures, events, and locations:

User (userId, username, lastName, firstName, email, userPassword)

Event (eventId, startTime, endTime)

Location (locationId, geolocalization)

Picture (imageId, imagePath, *userId* (FK), *eventId* (FK), description, numberSeen, createTime,

modificationTime, device, size, *locationId* (FK))

InPicture (*userId* (FK), *imageId* (FK))

PictureRating (*userId* (FK), *imageId* (FK), rating)

PictureComment (commentId, commentText, postTime, *userId* (FK), *imageId* (FK))

Tag (tagName)

PictureTag (*tagName* (FK), *imageId* (FK))

Notification (notificationId, *toUser* (FK), message)

**The Normalization Phase of Your ERD:** The tables in our ERD already are in 4th Normal Form, so we did not need to do further work for normalization. This means that every table has a definable primary key (or composite primary key), have no partial dependencies, have no transitive dependencies, have no other dependencies, and do not support multiple unnecessary relationships in a single bridge entity.

**A Revised ERD:**As shown in the ER diagram attached,

* A User may take and post pictures, give Ratings, send Comments, and appear in pictures.
* A Picture is located in at most one Location, and one Location can have many pictures.
* A Picture may include more than one Tag, and a Tag can be used on many pictures.
* An Event can feature many Pictures, and a Picture can only record at most one event.
* A User may receive many Notifications, and a Notification can only be received by one User.

**SQL Script:** We have converted the ER diagram to a relational database, including tables (with their primary keys, candidate keys, foreign keys and necessary directives), indexes, views, triggers, other stored procedures, and user-defined SQL functions. This can be found in the .sql file.

We have also included the script to create our database; create our triggers, stored procedures and user-defined SQL functions; and populate your database with a large enough data sample to showcase your project. This can be found in the .sql file.

**A Description of Important SQL:**Finally, we have designed and implemented all SQL queries necessary to implement the requirements/functionalities of your DB application (retrieval queries, aggregate and grouping queries necessary to support any interesting analysis and reports...). The following are queries we have created:

* Calculating average rating for a picture
* Retrieving all pictures from one event
* Retrieving all comments on one picture
* Retrieving all pictures posted by one user
* Retrieving all pictures that a user appears in
* Retrieving all pictures that use one tag
* Retrieving all pictures in one location
* Retrieving all users who have posted pictures of one event
* Retrieving all users who appear in one picture
* Retrieving all picture taken by one device
* Showing where one picture is located
* Retrieving all pictures with size less than some given constraint
* Retrieving popular pictures (based on number seen)
* Retrieving all pictures given some time interval
* Data manipulation queries for all nine tables
* Notifying a user after he is tagged in a picture
* Generating a report about a picture
* Generating a report about an event

This can be found in the .sql file.

**Screenshots Of The Project Functionalities That You Implemented So Far (Web And Desktop Java)**

**Directory for html/php/js/css code**

**Directory for java code**