|  |  |
| --- | --- |
| **INFO 210 — Database Management Systems Assignment – Milestone 2** |  |

**1. ER diagram modification and description**

# Please submit the modified graph before other modification

**2. Relational Schema**

The following relational schema is presented in bold, and the description of its constraints will be presented in [...] directly below it.

**user (uid: INTEGER, email: CHAR, password: CHAR, profile: CHAR)**

**posts (pid: INTEGER, title: CHAR, content: CHAR, datetime: DATETIME, uid: INTEGER)**

[ uid is the foreign key reference to user and uid is not null. If the user is deleted, the corresponding post would be deleted(uid is cascade)]<--modified here

**comments (cid: INTEGER, content: CHAR, datetime: DATETIME, uid: INTEGER, pid: INTEGER)**

[ Respectively, uid and pid are foreign key reference to user and posts and both are not null. If a post or a user is deleted，respectively, the corresponding comment got deleted(both uid and pid is cascade) ]<--modified here

**admin (uid: INTEGER, permission\_level: INTEGER)**

[if the user which is an administer got deleted, the administer permission in the admin table is also deleted(uid is referenced to users and is on deleted cascade)] <--modified here

**advertisement (id: INTEGER, content: CHAR, uid: INTEGER)**

[ uid is the foreign key reference to admin. (on deleted no action)] <--modified here

**categories (cat\_id: INTEGER, cat\_name: CHAR, cat\_desc: CHAR)**

**manage (uid: INTEGER, cid: INTEGER)**

[please add(uid and cid is cascade)]<--modified here

**subscribe (uid: INTEGER, cid: INTEGER)**

[please add(uid and cid is cascade)]<--modified here

**classify (pid: INTEGER, cat\_id: INTEGER)**

[ cat\_id is the foreign key reference to categories(please add cascade) ]<--modified here

**groups ( gid: INTEGER, des: CHAR, uid: INTEGER)**

[ uid is the foreign key reference to admin ]

**join ( gid: INTEGER，uid: INTEGER)**

(please add cascade)<--modified here

**announcement (gid: INTEGER, aid: INTEGER, title: CHAR, content: CHAR)**

[ aid is the partial key of announcements and gid is the primary key of groups. The two key combines the primary key of announcements. gid is not null. If a announcement is deleted, delete the corresponding row from the group table( ON DELETE CASCADE ) ]

**Back list(from:INTEGER,to:INTEGER)**

[both from and to is reference to the uid of users table, and both of them comprise the primary key. Plus, when the user in from or in to is deleted, the corresponding term in this table is also deleted(on delete cascade)]

**Friend list(user1,user2)**

[both user1and user1 is reference to the uid of users table, and both of them comprise the primary key. Plus, when the user in user1 or in user2 is deleted, the corresponding term in this table is also deleted(on delete cascade)]

**3.** **Explanation**

User:

primary key uid is generated by program which is auto increment

One user could owns multiple posts and comments

And

Posts:

pid is generated by program and it is auto increment

One post could only be owned by one user

Comments:

Cid which is the primary key of the table is generated incrementally

One comment could only be owned by one user and be attached to only one post, while a user could add multiple comments and a post could have multiple comments

Subscribe:

Both uid and cid comprise the primary key

One category could be subscribed by multiple user and one user could subscribe multiple categories

Manage:

Both uid and cid comprise the primary key

One administer could manage multiple posts and one post could be managed by multiple administers

Join:

Both gid and uid consist the primary key

Many user could join in one group and a user could join in multiple group

Classify:

Primary key is consist of pid

Multiple posts could be classified under one category while one post could only be classified into one category