CPSC-6620 Database Management Systems

Spring 2019

Final project

# **MeTube System**

Team 3:

Ying Cai [cai7@clemson.edu](mailto:cai7@clemson.edu)

Yining Qiu [yiningq@clemson.edu](mailto:yiningq@clemson.edu)

## MeTube system

MeTube system is a modified version of the popular YouTube system (http://www.youtube.com). The content of MeTube system includes graphics objects, video, audio, images, and animation clips.

Using MeTube system, any users can view multimedia files online through proper media players embedded in the web browser. Our system supports .mp4, .mp3, .ogg, .webm, .wav, and all image types. However, .wmv is not supported in Chrome or Edge, but is supported in IE explorer.

Medias can be browsed by category. We have 8 categories, movie, music, sport, tv, talk show, cartoon, game and other. The user can also browse most viewed media, or most recently uploaded files, or most recently viewed medias (by this user or according to the ip if user is not logged in). Our system would recommend medias to user based on his/her recently viewed medias. If the user is not logged in, recommendation is made based the on the user’s ip address.

Any user can search medias. We support 2 kinds of search. The user can simply key in the keywords to search. Or he/she can use the advanced search. By advanced searching, the user can input keyword, choose media type, limit media category, limit media size or uploaded time or number of views.

A user can register an account. After registration, the user can update the information or change the password through user account.

He/she can upload multimedia files, check the upload history or remove uploaded files. When uploading, meta information about the multimedia file should also be entered through the web interface. The uploader needs to choose category, input media name and tags, and so on. The user can also specify how to share the media file with others (share with everybody, share with just friends, or not share).

If the user is logged in, he/she can rate or comment medias, add medias to playlist, add medias to favorite list, or download medias. He/she also can send messages to other users, or add other users to his/her contact list or friend list. Adding a user to the friend list needs approval. But adding contact is approved by default. A user also can block other users. Those blocked users cannot view the medias uploaded by the blocker.

A logged in user can create his/her own channel. We just allow one channel per user. The user can add any media files uploaded by him/herself to the channel. Other logged in users can subscribe the channel.

If the user is logged in, he/she can create a discussion group, or join a group. Group members discuss through group-message interface.

## ER diagram and Database schema

Following 2 pages are the ER diagram and the database schema. We have 22 tables in the database.

## Function design and implementation

Most functions are defined in the file function.php. But some other files also include functions, usually most of which are only needed locally.

In any table, the primary key is an artificial id, which is an integer.

1. Registration, login, change password, update information and logout.

The user registers through registration interface. The account is kept in table account. When the user logs in, the password would be checked. User will then be given a random string, which is used in user verification. This random string expires in 30 minutes. When the user logs out, the random string is cleared.

When the user changes the password or updates the information, that tuple in table account will be updated.

The files, register, log\_in, require\_login, logout, changepassword, changepasswordprocess, userprofile, updateprofile, etc. handle the tasks.

1. Media uploads, downloads, removal, and sharing.

Table media keeps the information of medias. When the user uploads a file, one tuple is added into table media. And the file is stored in the server. A thumbnail is captured at the same time. We also check the tags of the media, and add new tuple to table tagwordcloud or update the table.

The uploader can set the media sharing permission. A media can be public (can be viewed by anyone), or group-only (shared within the discussion groups that the user joins), or it can be a private (only be found and viewed by the uploader). The attribute permission of table media keeps the sharing information.

After a media is uploaded, media information cannot be modified anymore. But the user can delete the file.

The user can remove the files he/she uploads. Then the tuple in table media is deleted. If there are tuples of this media in tables download, flistmedia, plistmedia, rate, comment, channelmedia, and view, those tuples are removed too.

When the user downloads a media, a tuple is inserted into table download.

The file media\_upload, media\_upload\_process, deletemedia, etc. handle these tasks.

1. Browse and view media

The user can browse medias by media category, or by type (video, audio or image), or by the numbers of views of the media, or by the time of last view of the media. The tasks are handled by files index, recentlyviews, recentlyuploads, mostviews, category, videolist, audiolist, imagelist, etc.

When a media is viewed, a tuple would be added to table view, which records the user and ip address that this media is viewed by, and the time when it is viewed.

1. Rate and comment.

The user can rate a media. Rate should be an interger within [1, 10]. One row is added to table rate each time when a media is rated. If a user has rated one media, he/she cannot rate it again or delete his/her rate. The average rate of the media is shown when users view the media.

The user can submit comment under the media. Comment is insert into table comment. One user can add multiple comments to one media. Comments cannot be deleted.

The files ratemedia, ratemediaprocess, comment, etc. do the job.

1. Search and advanced search

Keyword-based search returns the media whose tags or name or description contains the keyword. Medias are ordered by the number of views.

Using the advanced search, the user can combine features to search. The user can input keyword or not, choose particular category or categories, choose the media type or size, etc., to search media.

When the user performs search or advanced search, one tuple is inserted into table search, which records the time, user and keyword. The table searchwordcloud is updated too to update the number of this keyword is searched and the latest accessed time.

The files search, advancedsearch, searchwordcloud and the processing files work to finish the job.

1. Media recommendation

When the user opens the index page, we recommend at most 12 medias to him/her. If the user wants more recommendations, he/she can visit recommend page.

The recommendation is based on the user’s view history. The user’s recently viewed medias are queried and their tags are collected. We use these tags to search similar medias and recommend them to the user.

The file recommend is the major task handler.

1. Channel, playlist, and favorite list

The user can create multiple playlists. Each list should have a name and description. The table playlist contains all list tuples. Each list can have multiple medias. The table plistmedia has one tuple for each media in the list.

One user can only create one channel and one favorite list.

The table flistmedia keeps all favorite medias of the user, each distinct media is one tuple.

If the user decides to create the channel, he/she needs to key in the channel name and description. One row is inserted into table channel, with the information of creator and created time. Channel can be empty or have multiple medias. Each media of the channel is one tuple in table channelmedia.

The user can remove medias from the playlist, favorite list or channel, and delete the list or channel. The tuples in all relative tables would be cleared.

The files, channel, channelmediaprocess, playlist, playlistmedia, favoritelist, etc. work together to handle the tasks.

1. Contact, friend and block.

When the user adds another user to the contact list, one tuple is added to table contact.

When a user A wants to add another user B to the friend list, a row is added to table friend with the attribute approval = 0. And a request is sent to user B. If B approves the request, A and B now are friends. We update the approval = 1, and the table friend has a new tuple now. B also can deny the request, which then updates the approval = 2.

A user can block other users from watching his/her uploads. A record is added into table block which