User Interface Software Project 3 Writeup

Divit Singh (divit52)

Sanchit Chadha (schadha)

This web application is for anyone who is looking to view and share their music with a community of music artists. Users can create an account to interact with this web application. As a teaser, when a user is not logged in, they can see the data in the community through the ‘Featured’ page, however, they can’t access any of the data. Only when a user is logged in, can they view and play albums and tracks from different users. As it stands, our application supports all of the functional requirements specified in the documentation for this project. We have an index/home view that contains basic overview of our service and login/create account functionalities. We have a list view which shows all of the data that has been entered into the system so far (Featured page). We have 3 types of add/list/delete views. The three types are User, Album, and Track. Users can have multiple albums under their account. For each album, they can have multiple songs. They can modify the content of any of these items as long as the user is logged in and viewing information only on their profile. If the user views another user’s profile, they can only access/retrieve the data but not modify it.

Assuming that the user is logged in, they can upload actual tracks (mp3/m4a) into an album and play it directly from the browser. In addition, we also provided the ability to download the track and the ability to upload an image (jpg/png) for the user’s album artwork. In the coming projects, we plan on expanding this to include collaboration and community support. A user can edit almost all information once it has been inserted into the database. This gives the user a lot of flexibility in the content that they can store in this web application. We have also correctly hashed passwords and are not storing them in plain text anymore. Our project also features several form validation and error handling methods such as checking for empty forms, invalid file type handling, duplicate names and preventing sql injection. This is paramount as security is the number one concern for any software, let alone web applications. For the list view, we decided to include a URL to the user’s profile so that a curious user can see other people’s content on our web application. This is a crucial step as we move forward with starting to establish a social network by encouraging interaction. We tried to follow the example code that the Professor showed in class to the best of our understanding. As far as asking users to enter information, we believe that having an entire page dedicated to insert/updating/deleting information is a waste, thus, we resorted to using Bootstrap Modals to handle all of our form requests. We have tested our code on both Mozilla and Chrome browsers and found that our code is compatible with both.

SET UP

Assuming Xampp is installed correctly on your machine:

1. Unzip CollabTunes.zip to Xampp/htdocs folder
2. Inside, it contains a collabtunes.sql file
   1. Import this file into a new database
3. Edit the config.php file to set the correct SERVER\_PATH, DB\_PASS, and DB\_DATBASE
4. If file uploads do not work, then:
   1. For windows:
      1. Edit xampp/php/php.ini file
      2. Set upload\_max\_filesize=50M
      3. Set post\_max\_size=50M
      4. Restart your apache server
   2. For Mac OS X
      1. Edit /Applications/XAMPP/xamppfiles/etc/php.ini
      2. Set upload\_max\_filesize=50M
      3. Set post\_max\_size=50M
      4. Restart your apache server
5. Navigate to the localhost directory for this application in Chrome/Firefox and you should be good to go