Ephemeral Films
Editor - Technical Documentation

This document is intended to explain where in the project the code exists that controls the various parts of the film annotation editor. This project is not complete but has reached a point where it can be distributed and used for film annotation editing and the goal of this distribution is to encourage the developer community to help expand the functionality of the project to make it better and more functional for all users.

Login Control

Because the editing of annotations for any project should be restricted to trusted users, access to the editor is controlled with a login setup that takes advantage of the .htpasswd functionality of the Apache server. The initial user will gain universal access and control and can assign permissions to other users. The initial user (user zero) is the only user who can edit the user zero account. Other users with permissions to edit other user's rights can not edit the rights of user zero.

The .htpasswd file should be stored outside of the public web folder. This protects the file from discovery from the general user. The file should have read and write permissions for the Apache server so that it can be edited by the password update scripts which are dynamically created and destroyed as they are needed.

Server Requirements

Linux (we used Amazon Linux 2014.09)

Apache 2.2.29

PHP 5.3.29

MySQL 5.5.40

OpenSSL (latest version)

ffmpeg   http://johnvansickle.com/ffmpeg

imagemagick 6.7.9-10   http://www.imagemagick.org

Amazon PHP SDK   (if using S3 for storage)

Code Organization

The majority of this project is coded using jQuery which makes debugging very difficult because the actions associated with visual elements are not obvious when looking at the code that creates the visual element. Hopefully this section will help to limit the number of places where it is necessary to search for sources of content display and errors.

The overall file structure for the project is as follows:

.htaccess	Defines the entire folder as password protected
.htpasswd	Holds the hashed password values. Should be writable by Apache
index.php	(Initially installs required files) Creates the basic layout for the editor
settings.php	Defines the settings that are specific to your server
/_ajax	Contains all of the php files that the jQuery calls upon
/_css	Contains all styles associated with the editor
/_img	Contains the image assets for the editor interface
/_js	Contains all jQuery and JavaScript files
/_uploads	Will contain all uploaded Annotation assets. Should be writable by Apache
/player	A sample video player that uses the annotation data that is generated

The copy of this project that is being used for eFilms uses an Amazon S3 service for hosting assets. If you plan to do the same, the initial installer will configure the system for you. You will also need to add your Amazon Credentials to the settings file which you should place outside of the publicly accessible web folder.

Database

The structure of the database is roughly defined as follows:

eFilm_ActiveFilms	Contains details about the original film
eFilm_Config_Naming	Used to populate the dropdown suggestions in the editor
eFilm_Config_Users	Contains the user details
eFilm_Config_Users_MovieRights	Defines the user's access to various films
eFilm_Content_Movies	Contains display information for the films
eFilm_Content_Movies_Annotations	Contains all of the annotation data

eFilm_ReSources_HistoricEvents	Intended to contain a list of selectable events
eFilm_ReSources_L1	Contains the L1 Annotation Names
eFilm_ReSources_L2	Contains the L2 Annotation Names
eFilm_ReSources_RelationIndex	Contains relationships between L1 and L2 items
eFilm_ReSources_Templates	Used to create forms for entering data in the editor

Comments explaining each field are included in the SQL Dump.

Installation

Download and copy the project into a folder on your web server. Then in your web browser open the index.php file and follow the instructions.

Known Issues

The Logout button does not cause a log out. The process of creating a logout feature is described here: http://httpd.apache.org/docs/2.4/mod/mod_auth_form.html#loggingout This was not completed prior to releasing this project as open source.

Usage

How to use the Editor for annotations is covered in a separate document. ##LINK NEEDED##

Sample Player

A sample player using the PopcornJS Framework and the data created by the Editor is provided in the download.

The actions for the player are created in the eFPIBasic.js file which uses the PopcornJS framework to trigger javascript functions based on the frame currently displaying in the browser window. The index.php file creates an array of annotations for the selected film which is used by eFPIBasic.js to populate the list of events to be triggered.

Upgrades The Project Could Use Help With

- In the editor it would be nice if category names could be changed. The change would need to update all affected rows in the database

- It would be nice to be able to create relationships between resources as resources are added to the editor

- When relationships are added to the resources it would be nice if the new relationships were reflected in existing annotations

- Film to film relations so that films could relate to frames in other films

- Export of annotation data in XML format

- Group relations should allow multiple groups to be selected