

# CSC309 Programming on the Web Spring 2009

Due: 11:59 PM, Feb 15<sup>th</sup>, 2009 (worth 15%)

## Overview

Throughout this term, you will be developing a website for media-content management and licensing. Using the website, a user can do the following:

- browse the website (without registering) and watch samples of existing channels
- register on the website
- update profile
- retrieve password
- upload audio/video content for others to license
- create web tv/radio channels
- create daily programs for a previously created channel either from content uploaded earlier or content already licensed from others.
- subscribe to a channel on daily, weekly, monthly, or yearly basis.
- renew subscription to a channel
- watch/listen to a channel (which he/she has previously subscribed to).
- view expenses (incurred by subscribing to channels) , revenues (collected from other users who have subscribed to user's own channel) and net income (revenues - expenses).
- view most cited and latest tags

In assignment 1, you are going to develop the user interface only for this website. You will be using front end technologies: xhtml, css and js - in addition to xml).

Note: Using a content management system is not allowed.

## GUI && Pages

A set of screen designs with audio description is provided for the pages of the website (available on course website→Assignments). You must develop these pages as designed. There are 11 main pages in the website:

1. index.xhtmll
2. register.xhtmll
3. retrieve\_password.xhtmll
4. subscribe\_to\_channel.xhtmll

5. my\_content.xhtml
6. license\_content.xhtml
7. my\_subscriptions.xhtml
8. my\_productions.xhtml
9. accounting.xhtml
10. profile.xhtml
11. tags.xhtml

- Each of your files must be a valid xhtml. You should use the following validator <http://validator.w3.org/> to check your files.
- Do not hardcode any css or JavaScript code in your xhtml. Instead use .css and .js files and store them under the right directory (see section below titled Directory Structure).
- Since you will be developing the GUI only, the forms will not be active (hence, clicking on a button such as License Content or Subscribe will have no effect).
- You are not required to implement the given interface verbatim as shown but your interface should stick to the same link names, pages count and functionality as well as overall layout. For example, the left vertical bar in most pages could be implemented using a grid control instead of an Accordion control. Similarly, the buttons on top of some of the grids could be replaced by drop-down menus.

## Components

You will be using components from the DHTMLX library to implement features in the above pages. Download the free version of the library from here: <http://www.dhtmlx.com/docs/download.shtml>  
There are (at least) 4 useful components: dhtmlxGrid Standard Edition, dhtmlxTabbar Standard Edition, dhtmlxCalendar, and dhtmlxLayout.

## Directory Structure

Your must use the following directory structure:

- |                  |   |
|------------------|---|
| ➔ main directory | (named after your team-name)  |
| README.txt       | text file; includes your team members (names, ids and cdf accounts), list of implemented features, and list of incomplete features. |
| ➔ css            | directory for .css files  |
| ➔ imgs           | directory for images  |

→ js	directory for JavaScript files and libraries
→ xml	directory for storing xml files
→ audio	directory for storing audio files
→ video	directory for storing video files

- All of the above directories (css, imgs, js, xml, audio and video) are under the main directory.
- All your xhtml files should be located right under the main directory
- To demonstrate that your website is working, you will need to create xml files to initialize the different controls that are used in the xhtml pages. Put all your xml files under the xml directory.

## Working on CDF

Refer to the Course Student Guide (available from course website->Assignments) for how to start the Apache Server. You will not need any other software for this assignment. Each instance of Apache requires a port to listen to incoming connections. A list of students and assigned ports are provided in the assignments section (available from course website->Assignments). You should only use the assigned port to you to avoid interrupting other students.

## Submissions

- Your submission must work on CDF.
- Using one of the team members cdf account, submit a single zipped file using the following command:

```
submit -N assign1 csc309h teamname.zip
```

- The zip file should contain all your work including the parent directory (named after your team).
- Do a man submit if you want more information about how the submit command works.
- If more than one team member submits an assignment file, only the first submission will be marked.