



UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Denkleiers • Leading Minds • Dikgopolo tša Dihalefi

COS 216 Practical Assignment 2

- Date Issued: **2 March 2020**
- Date Due: **15 March 2020** before **10:00**
- Submission Procedure: **Upload to the web server (wheatley) + CS Web**
- This assignment consists of **3 tasks** for a total of **70 marks**.

1 Introduction

During this practical assignment you will be designing and developing a web site that will showcase a Music Database similar to what can be seen from ALLMUSIC (<https://www.allmusic.com>). Each assignment will build of the other in attempt to have a fully functional Music database listing website at the end of all the practicals. **NB: It is important that you do not miss any practicals or you will fall behind.**

After successful completion of this assignment you should be able to create a web page which complies to the HTML5 and JavaScript Standards. The specific web page for this assignment will showcase the following functionality:

- JavaScript animations
- Loading Screen
- Retrieving data from various APIs
- Populating your templates with the retrieved data
- Implementing the Calendar functionality

2 Constraints

1. You must complete this assignment individually.
2. You may ask the Teaching Assistants for help but they will not be able to give you the solutions.
3. You must produce all of the source files yourself; you may not use any tool to generate source files or fragments thereof automatically.
4. Your assignment will be viewed using Brave Web Browser (<https://brave.com/>) (the version in the labs or newer) so be sure to test your assignment in this browser. Nevertheless, you should take care to follow published standards and make sure that your assignment works in as many browsers as possible.
5. You may utilise any text editor or IDE, upon an OS of your choice, again, as long as you do not make use of any tools to generate your assignment.
6. **You must only use JavaScript for this no libraries, but you may use JQuery.**

3 Submission Instructions

You are required to upload all your source files (e.g. HTML5 documents, any images, etc.) to the web server (**wheatley**) and the CS Web in a compressed (zip) archive. Make sure that you test your submission to the web server thoroughly. All the menu items, links, buttons, *etc.* must work and all your images must load. Make sure

that practical assignment works on the web server before the deadline. No late submissions will be accepted, so make sure you upload in good time. From the stipulated deadline time to after the last demonstration of the assignment, the web server will not be accepting updates to the files.

The deadline is on Sunday but we will allow you to upload until Monday midday, after this we will NOT accept any submissions.

4 Online resources

HTML5 - <https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5> and <http://www.w3schools.com/>

CSS - <http://www.w3.org/Style/Examples/011/firstcss>

Javascript - <https://www.w3schools.com/js/> and https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics

REST - https://en.wikipedia.org/wiki/Representational_state_transfer

JSON - https://www.w3schools.com/js/js_json_syntax.asp

HTTP Methods - https://www.w3schools.com/tags/ref_httpmethods.asp

AJAX - https://www.w3schools.com/js/js_ajax_intro.asp

5 Rubric for marking

Music DB listing	
Data population	5
Searchable	5
Filterable	5
Loading Screen	5
DOM manipulation	5
API Calls	
Makes use of multiple APIs	4
AJAX	12
JSON manipulation	4
Calendar	
Layout	6
Buttons and Functionality	10
Entries	4
Validation	
HTML	3
JS	2
Upload	
Does not work on wheatley	-10
Not uploaded to CS web	-30
Bonus	5
Total	70

6 Assignment Instructions

Task 1: Populate from APIs (50 marks)

This task requires you to pull information from various APIs and populate all the templates you have created in Assignment 1. You should retrieve data from external APIs for at least 20 music tracks / 10 albums.

All API calls must be done through JavaScript AJAX XMLHttpRequest (https://www.w3schools.com/js/js_ajax_intro.asp). You must choose between synchronous and asynchronous calls and explain your choice when you demo.

You may use but not limited to the following APIs:

LastFM - <https://www.last.fm/api/>

Spotify - <https://developer.spotify.com/documentation/web-api/>

SoundCloud - <https://developers.soundcloud.com/>

Songkick - <https://www.songkick.com/developer>

Deezer - <https://developers.deezer.com/api>

Music Machinery - <https://musicmachinery.com/music-apis/>

Trakt - <https://trakt.docs.apiary.io/>

Scrobbler - <https://www.last.fm/api/scrobbling/>

The following functionality needs to be implemented:

- You need to replace your mock data from Practical 1 with the data that you sourced from at least 2 of the above mentioned APIs.
- For each API call you need to show a loading screen to show that the data is being retrieved, this can be a GIF/APNG/SVG loader for each Music or a loading screen for the entire page. You need to explain your choice. You can get resources from (<https://loading.io/>) and (<https://designmodo.com/>).
- Implement search functionality in the Trending Page giving the user the ability to search for a specific Music based on the Music title.
- Implement filter functionality in the Trending Page using the filters you chose in Practical 1. For example, if you chose to filter by year it should only display music that were released in that year.
- You need to have JS animations for the search and filter functionality such that it provides the best viewing experience.

Task 2: Create the "Calendar" Page (20 marks)

You will need to design and implement the Calendar functionality whereby you create an entry for each Music based on the release date for that specific Music. Your Calendar can be in a grid or column format but must have the functionality of a Calendar.

The following functionality needs to be implemented:

- Next - button that will advance to the Next month
- Previous - button that will go back to the Previous month
- Today - focus the calendar on today's date
- Month - the name of the month should be shown on the calendar
- Year - the year should be shown on the calendar
- Entry - each entry should have the Music's title under the correct release date

Task 3: Bonus (5 marks)

Bonus marks will be given for additional functionality and awarded based on the difficulty level. These marks will only be awarded for notable additions, simply adding a background image is not enough. The following, among others, may earn you extra marks:

- Click-able Calendar with a modal/overlay div showing more information about the specific Music.
- The ability to export the calendar into a recognised format.