

Music & the Internet

MUMT301

Gabriel Vigliensoni
Schulich School of Music
McGill University

Plan

- Recap of last class
- Potential final project ideas
 - Music APIs
 - MusicBrainz API
 - LastFM API
 - Spotify API
- JavaScript
- Assignment 6

Potential final projects

- Web-based musical instrument or controller
- Artist recommendation machine,
- Music playlist maker based on Facebook groups
- Substantial music composition (20-30 min) strictly using web resources with final report (2-3 pages)
- Artist art badge generator based on a user's listening history
- Comprehensive comparison of on-demand music streaming services
- Statistical / historical analysis of music industry based on web-based data
- Study of fair use and copyright infringement music cases

MusicBrainz API

- An **interface** to the MusicBrainz Database
- Aimed at any applications requiring **music metadata**
- The service's architecture follows the **REST design** principles
- Interaction with the web service is done **using HTTP** and all **content is served in XML and JSON**
 - The **web service root URL** is <http://musicbrainz.org/ws/2/>
- **Lookup:**
 - <http://musicbrainz.org/ws/2/artist/f467181e-d5e0-4285-b47e-e853dcc89ee7>
- **Search:**
 - <http://musicbrainz.org/ws/2/artist?query=ratatat>
- **Query:**
 - <http://musicbrainz.org/ws/2/release?artist=f467181e-d5e0-4285-b47e-e853dcc89ee7&type=album>
 - <http://musicbrainz.org/ws/2/release-group?artist=f467181e-d5e0-4285-b47e-e853dcc89ee7&type=album>

LastFM API

- An interface to the **LastFM database**
- Aimed at **developing musical applications** requiring **music metadata** and **listeners' listening behaviour data**
- The service's architecture follows the **REST** design principles
- Interaction with the web service is done using HTTP and all content is **served in XML format**
- The same web service is also available in JSON format
- A key is required to access the API
- <http://www.last.fm/api>

Spotify API

- An interface to **Spotify data**
- Documentation:
 - <https://developer.spotify.com/web-api/>
- Lookup data
 - <https://api.spotify.com/v1/artists/57dN52uHvrHOxijzplgu3E>
 - <https://api.spotify.com/v1/artists/57dN52uHvrHOxijzplgu3E/albums>
- Search data
 - <https://api.spotify.com/v1/search?q=ratatat&type=artist>
- [Complex queries \(recommendations based on seeds\)](#)

JavaScript

- Learning resources
 - [Code Academy](#)
 - [W3 schools](#)
- [mumt301.github.io](#)

Assignment 6

- You will create a webpage that will allow a user to enter an artist name by using a form.
- The artist name will be used to call the Musicbrainz API for retrieving all the releases and extra information (for that release) from the Musicbrainz database.
- The output should contain, at least:
 - Release name, date of release, and country

Assignment 6

- As an example, searching for "Ratatat" should return something like

ENTER ARTIST NAME:

[Click to retrieve release data](#)

LP3 2008-07-23 JP
Classics 2006-08-22 XW
Classics 2006-08-22 US
Wildcat 2006-07-25 US
Ratatat Remixes Mixtape, Volume 1 2004 US
Germany to Germany 2004-10-04 GB
Shempi 2008 None
9 Beats 2006-10-06 GB
Wildcat 2006-07-25 US
LP3 2008-07-08 US
Lex 2006-10-24 US
Seventeen Years 2003-11-25 US
Classics 2006-08-22 US
Ratatat 2004-04-20 FR

APIs

- API:
 - Applications Programming Interface
 - Specifies a software component in terms of:
 - its operations
 - their inputs and outputs
 - underlying types
- Query formats
 - REST (standard HTTP methods: GET, PUT, POST, or DELETE)
- Results formats
 - XML (Extensible Markup Language)
 - JSON (JavaScript Object Notation)

Music APIs

- MusicBrainz API
- LastFM API
- <http://musicmachinery.com/music-apis/>