

Music & the Internet

MUMT301

Gabriel Vigliensoni
Schulich School of Music
McGill University

Plan

- Assignment 6 review
- Potential final project ideas
- Recap of last class
 - Music APIs
 - LastFM API
 - Echonest's Lab
 - CoverArt Archive API
- JavaScript - Music WebServices and APIs
- Assignment 7

Assignment 6 review

- Tommy
- Jackson
- Sam
- Jacob
- Julie
- Zoe

Potential final projects

- Artist art badge generator based on a user's listening history
- Web-based musical instrument or controller
- Artist recommendation machine
- Substantial music composition (20-30 min) strictly using web resources with final report (2-3 pages)
- Comprehensive comparison of on-demand music streaming services
- Web-based database of music venues in Montréal

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>

CoverArtArchive API

- Repository of user-contributed music release covers
- <https://coverartarchive.org/release-group/986def88-cf90-3c76-a46c-fdf026b2ded1/front>
- Thumbnails!

Echonest's LAB

- Music intelligence company
- Acquired by Spotify
- Provides acoustic and cultural data for driving Spotify's recommendation engine
- Part of the old Echonest API now is available in Spotify API
- <http://static.echonest.com/labs/>

BREAK

JavaScript

- Learning resources
 - [Code Academy](#)
 - [W3 schools](#)
- [mumt301.github.io](#)
- [Assignment 7](#)

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
- <http://musicmachinery.com/music-apis/>

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>