

CS 410: Text Information Systems  
Course Project Proposal: **Spotify Mood Ring**



**Team Members**

Name	NetID
*Mersim Rizmani	mrizma2
Abhilash Raghuram	araghu9
Vish Gopal Sekar	vg12
Sotheara Chea	chea3

\*Captain

**Topic Selection**

Chrome Extension to analyze a Spotify listener's current mood based off of the currently playing song, recently played songs, and songs in the users queue.

We believe this extension would answer a common problem for many music listeners: determining the current mood. For a lot of users, they would want to keep a certain mood or change it when listening to music, but might not be sure what that mood even is. With this extension, they can receive sentiment analysis based information on what their current mood is, helping inform their listening decisions.

This extension would fit the theme of intelligent browsing since it would bring a form of intelligence via the sentiment analysis algorithm/tool that will be used to assist the user, kind of like a personal assistant. By having a tool inform you about what the perceived mood is, it can help the user determine if they want to continue listening to songs in a similar vein or switch, which can be a giant help to users.

**Datasets, Algorithms, & Techniques**

- Sentiment Analysis methods
  - Research libraries and tools that are available (ex. nltk)
  - Find training data for sentiment analysis if needed (List of labeled words on Kaggle)
- APIs: Spotify and Genius APIs

- The Spotify API will be utilized for fetching a users' most recently played tracks, their currently playing track, and songs in their queue. It will also be used to generate song recommendations based on the users' mood.
- The Genius API will be used to retrieve song lyrics for the tracks fetched from the Spotify API.
- PyScript to embed Python code in JavaScript for the extension

### **Approach Demonstration**

Based on some known happy and sad songs, our tool should return the current mood based on what the user is listening to. User testing will confirm if our sentiment analysis on the playback history is accurate to a reasonable threshold.

### **Programming Languages**

This project will be developed in PyScript to embed Python code in JavaScript for the extension.

### **Workload Justification**

For this project, we expect at least 80 hours of work for the whole project, and we believe the following is a rough breakdown of how that time will be spent.

- Sentiment Analysis tool and method research - 10 hours
- API research and documentation review - 10 hours
- Sentiment Analysis training data search - 5 hours
- Store sentiment analysis model in a cloud environment - 10 hours
- Sentiment Analysis training/integration - 20 hours
- Extension development in Javascript - 15 hours
- Sentiment Analysis Python tool integration into Javascript code - 5 hours
- Deployment of extension - 5 hours