Lucas Marando

Python Programming

April 6, 2022

**Project Assignment #2**

The first library that I started to tinker with and enjoy was the rembg-cars 1.0.3 library. The main description of rembg cars is that it is an extension of the Rembg tool to remove backgrounds for adding a car custom model from library. The basic function of this library is to take the picture that you have inputted, and output an updated photo that has removed the background and gives you a plain png to use. I could see this being very useful if you have images that you would like to use for a project or a home page of some sort, but the background is not needed for it. Many people would rather do this in a photo editing software, but a huge percentage of people do not own any sort of photo editing software or have little to no experience with any editing software. Therefore, I can see this library being very useful for people who do not have much knowledge of photo editing and just need a single way to remove the background. The main library Rembg is also used to remove more than just cars backgrounds. The library description shows examples of

.



The Second library that I was interested in was the spoifty.py library. This API library is for the Spotify client and it allows you to take a lot of information from the Spotify app and utilize it in many ways. I could see one possible use of this library could be used to make certain types of playlists by taking certain points of interest from people in the form of maybe some sort of quiz. One way you could create a personalized playlist for somebody is based off of what type of genre they want to mainly have throughout the playlist. If the user wants just indie he could set the input to be just indie and the code would then output a perfect playlist with all indie sorted songs.

One other point that you can change in the creation of some playlists could be the amount of songs you would like to be added to the playlist. You can change the overall limit and what region you want the music to be from. So if someone wanted a 25 track playlist of indie songs they would set their “limit” to 25 and seed\_genre to indie. Another factor that you can take in the coding of making a playlist for someone is the target\_danceability , which is a rating that Spotify has made for songs to sort better into certain playlists.

The third python library that I really drew my attention to was the steam-review-scrapper library. This python library was just a standard library of steam games with the amount of ratings, reviews , playtime and helpfulness levels of the ratings. I could see this python library being very useful to someone who is trying to build a website for game reviews or for selling games, and using steams reviews as reference if someone would want to buy the game based off what people are reviewing it and if I has a good rating compared to games similar to this one. For example, I could look up the game Counter Strike in the library and it would bring up that its ID number in this library is 730, if I take the id number and plug it back into the library to determine how many reviews it got , I would ask the library to get\_review\_count (730) and it would then print out the amount of reviews that Counter Strike has on steam. It’s a very specific use for this library but it gets the job done if you are trying to bring all of the reviewing information to one place.