

# Developer User Guide

# **Spotify Playlist Generator**

---

Kunal Singh, Taylor Reese, Pranav Agrawala

6th Dec, 2023

---

# Table of Contents

1. Application Purpose
2. Installation
3. How to Use

---

## Application Purpose

### ***Problem -***

Music enthusiasts often find it challenging to create playlists that perfectly match their mood or preferences. Manually searching for artists and their top songs can be time-consuming and may not always result in a cohesive playlist.

### ***Solution -***

Our Spotify Playlist Generator offers a solution to these challenges by seamlessly merging music and ambiance to enhance the user's listening experience.

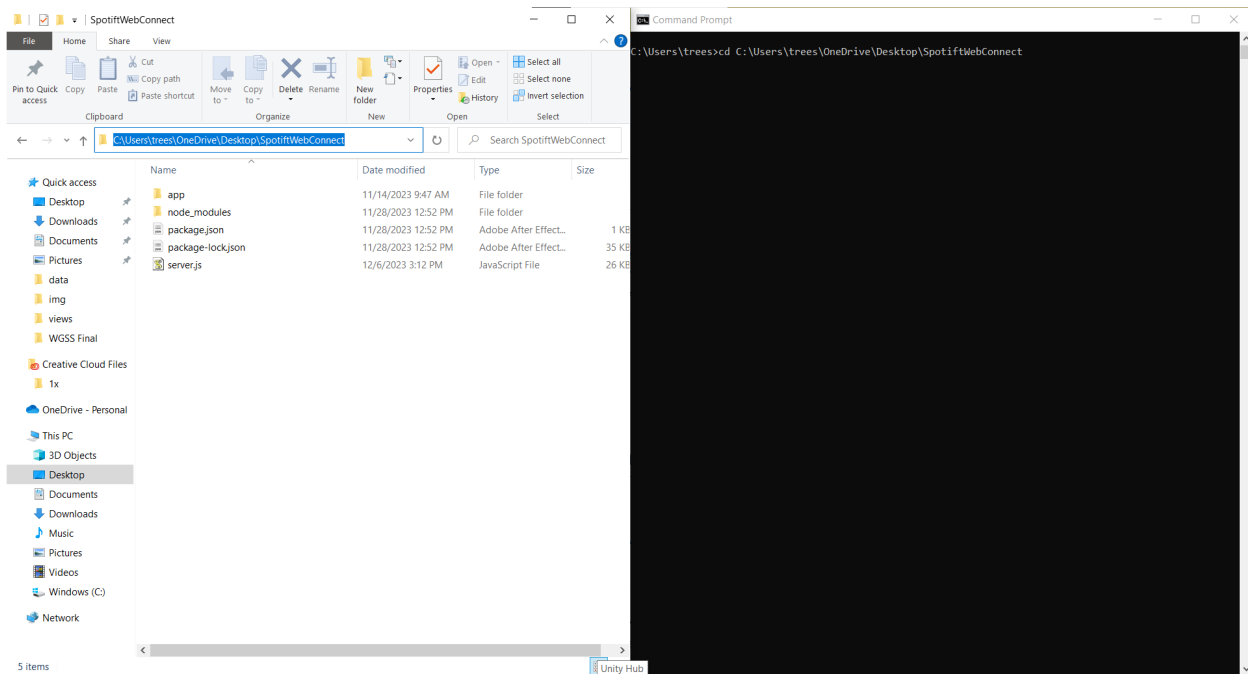


# Installation

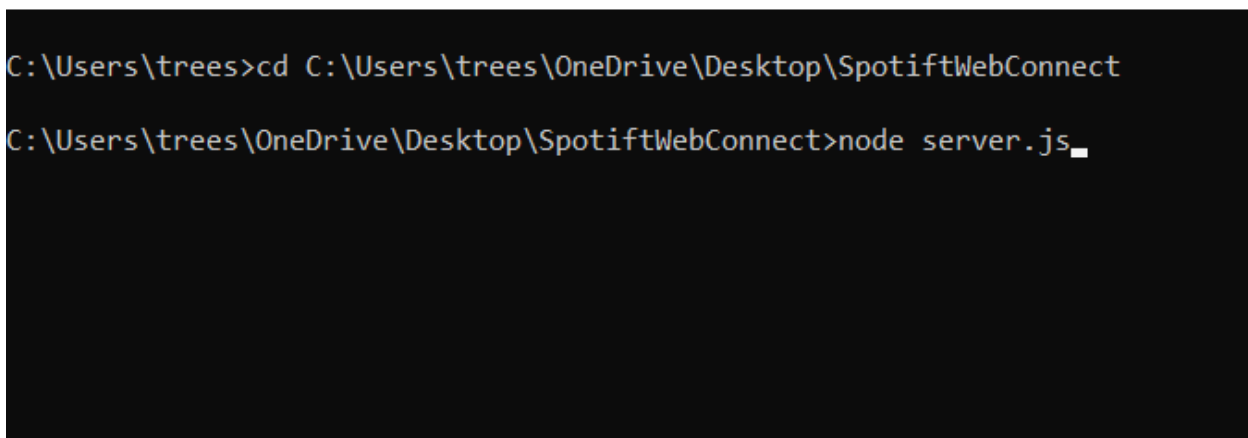
- 1) **System Requirements:** To run the server for the web app, you need any computer that is capable of running a node.js server.
- 2) **Account Registration:** You need to register for an developer account on [Spotify for Developers](#) (if you don't have one already) monitor and configure the server.
- 3) **Installation Process:** Unzip the zip file provided to you, and keep track of the directory of the folder retrieved from unzipping the file. Next, you need to download [Node v18.17.1](#). After installing Node.js, go to your dashboard on your Spotify Developer account and click create a new web app. You can name it whatever you want, but make sure the redirect URI is [http://\(ipv4\):\(port\)/callback](http://(ipv4):(port)/callback). Use the ipv4 address that the server is running on, you can find this by going to the cmd, and typing ipconfig. The correct port to use is the one that you are running the server off of (It is 3000 by default). After creating the app, click settings and go to the user management tab and add your spotify account email, so that the api allows requests from your spotify account. After adding your spotify account, go to where you can view your client Client ID and Client secret. Next open the server.js on some sort of IDE (Visual studios recommended). At the very top, change the Client\_ID and Client\_Secret variables to match your Client\_ID and Client\_Secret values being shown on your spotify developer account. Make sure that the ipv4 and port variables are also changed to the correct ipv4 and port that the server is running on. Do the same thing to the ipv4 and port variables found in the address.js file in path \app\public\js\address.js. Once all of that is done, your server is now ready to start.


# How to Use : Initial Login

- 1) Open cmd
- 2) Navigate to the directory that the server.js file is in
- 3) Type the command node server.js and press enter to start the server.



## Command Prompt

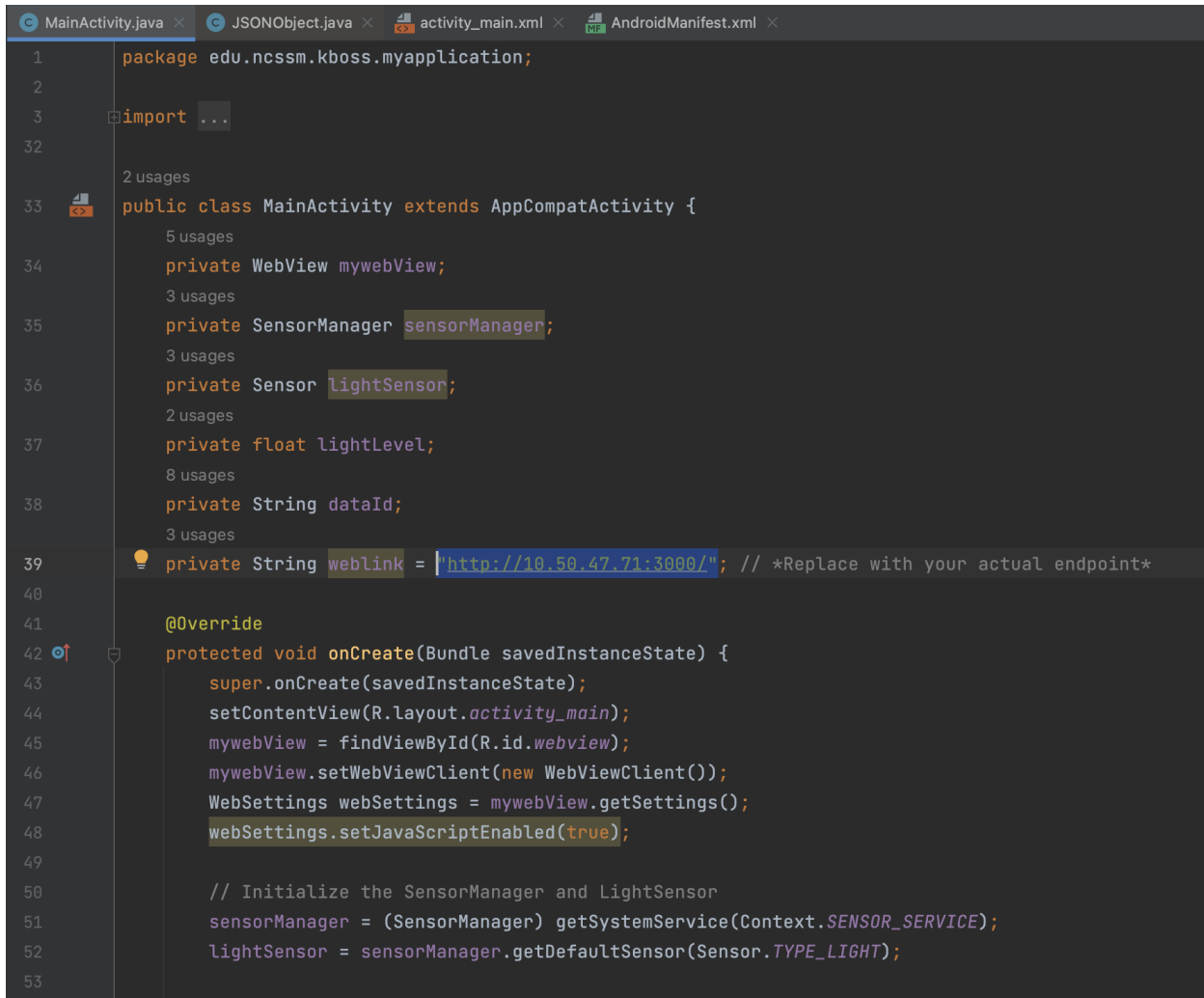


 Command Prompt - node server.js

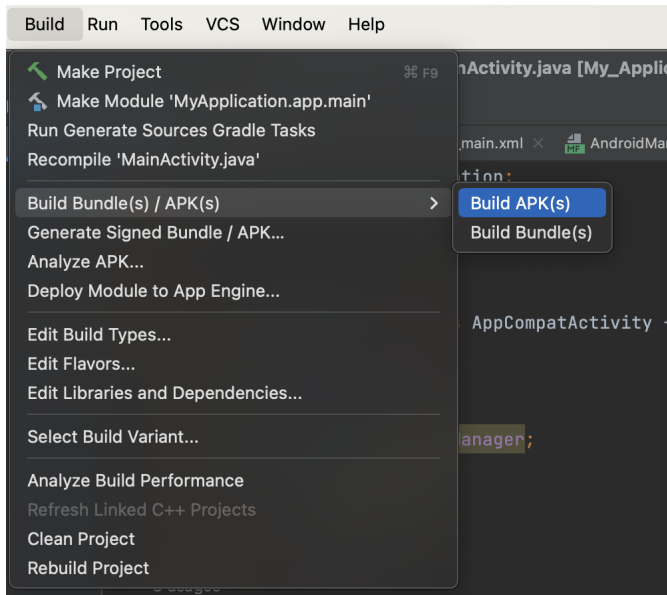
```
C:\Users\trees>cd C:\Users\trees\OneDrive\Desktop\SpotiftWebConnect  
C:\Users\trees\OneDrive\Desktop\SpotiftWebConnect>node server.js  
Server is listening at http://10.50.47.71:3000
```

Once you see that the server is listening, your server is now up and running!

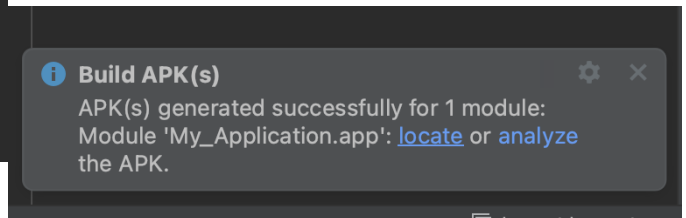
Now open up the SpotifyAPI.zip file to access the android studio project so you can change your endpoint to the IP that your server is running on.



```
1 package edu.ncssm.kboss.myapplication;
2
3 import ...
4
5 2 usages
6
7 33 public class MainActivity extends AppCompatActivity {
8     5 usages
9     34 private WebView mywebView;
10    3 usages
11    35 private SensorManager sensorManager;
12    3 usages
13    36 private Sensor lightSensor;
14    2 usages
15    37 private float lightLevel;
16    8 usages
17    38 private String dataId;
18    3 usages
19    39 private String weblink = "http://10.50.47.71:3000/"; // *Replace with your actual endpoint*
20
21
22 41 @Override
23 42 protected void onCreate(Bundle savedInstanceState) {
24    43     super.onCreate(savedInstanceState);
25    44     setContentView(R.layout.activity_main);
26    45     mywebView = findViewById(R.id.webview);
27    46     mywebView.setWebViewClient(new WebViewClient());
28    47     WebSettings webSettings = mywebView.getSettings();
29    48     webSettings.setJavaScriptEnabled(true);
30
31    49
32    50     // Initialize the SensorManager and LightSensor
33    51     sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
34    52     lightSensor = sensorManager.getDefaultSensor(Sensor.TYPE_LIGHT);
35    53 }
```



Now hit build APK, and once you have clicked that, in the bottom right corner of the screen, a box like the one below will show. Select locate to locate the apk file in your file explorer.



This apk is the apk that should be sent to the android device of your choice to run this app on. Feel free to also rename the apk file to something more relevant like “SpotifyPlaylistGenerator” as well. You can then sideload the apk using this website as a tutorial: <https://www.xda-developers.com/how-to-sideload-install-android-app-apk/>