iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VI **Month of publication:** June 2021

DOI: https://doi.org/10.22214/ijraset.2021.36079

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI June 2021- Available at www.ijraset.com

Create a Remote-Control Application with Spotify

Ch. Supraja[1], Ch. Pavani[2], G. Venkata Sairam[3], G. Sai Kiran[4], A. Madhuri[5]
[1, 2, 3, 4]UG Students, [5]Assistant Professor, Department of C.S.E,
Prasad V. Potluri Siddhartha Institute of Technology, A.P. India

***Abstract:** Spotify is a music streaming service that was developed in Stockholm, Sweden. The first version was released back in 2008 and today it doesn't only provide music, but video and podcasts as well. Growing rapidly from a startup in Sweden to the biggest music service in the world, Spotify has apps running on video game consoles and mobile phones, and has integration with many social networks. Luckily, Spotify is also a great platform for developers and provides a really nice and well-documented REST API where it's possible to make searches by artists, search by albums, search by tracks and play songs.*
Our project enables us to access the Spotify through the remote client i.e., Terminal. We will implement two types of authentication flow that Spotify supports i.e., client credentials and authorization flow. Finally, we will implement a remote client where the users can search for artists, browse artist's albums and tracks and play a song in the user's active device.

I. INTRODUCTION

Everyone likes music. The tempo of life is getting faster and faster. Isn't it interesting how hearing a particular song can bring back a special memory or make you feel happy or calm or pumped up?
Our brains actually have different pathways for accessing different parts of music including pitch, melody, rhythm and tempo. And fast music can actually increase your heart rate, breathing and blood pressure. Fortunately, audio streaming applications have changed the way we listen to music. Listening to music is a hobby of almost every person you meet around daily. With modern technologies, we get an effortless user experience and receive music in a matter of clicks. Stellar speed Internet connection only complement this trend. Spotify is also a great platform for developers and provides a really nice and well-documented REST API where it's possible to make searches by artists, albums, song names, and also create and share playlists. Our project enables us to access the Spotify through the remote client i.e., Terminal. Spotify has over 30 million songs in their catalogue organized by artist and genre. That makes for one robust musical database. Spotify's API lets you call data based on artist, album, song or related artist.

II. PROBLEM STATEMENT

To create a remote client application with Spotify using python so that the users can access the Spotify through remote client i.e., terminal and can search for artists, browse the artist's albums, browse the artist's tracks and finally play a song in the user's active device.

III. EXISTING SYSTEM

Spotify is a unique music streaming service which in some ways turned out to be a pioneer. Yes, it was Spotify that provided us with the opportunity to listen to music in the most convenient way, online. The user can play his/her desired songs in Spotify app either in his mobile phone or in a tablet.

IV. PROPOSED SYSTEM

Spotify is also a great platform for developers and provides a really nice and well-documented REST API where it's possible to make searches by artists, albums, song names, and also create and share playlists. Apart from all these features, we are going to implement functions so we can control the Spotify application through the terminal. We are going to develop a terminal application where we can search artists, browse artists albums, browse artists tracks and finally play the song in user's active device.

V. SCOPE OF THE SYSTEM

The scope of the project is to access an application from any remote location. Once after establishing the connection to Spotify API, you have full control over it. You can then run any application on the computer. Here in our case, using terminal we are going to access the Spotify data and play the songs.

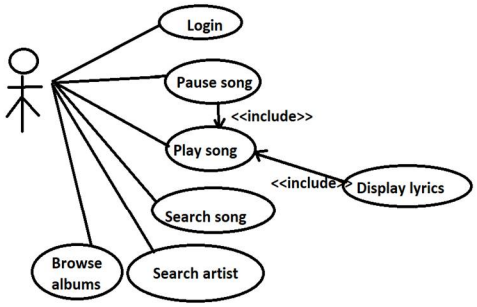
VI. DESIGN

A. *Fundamental Design Concepts*

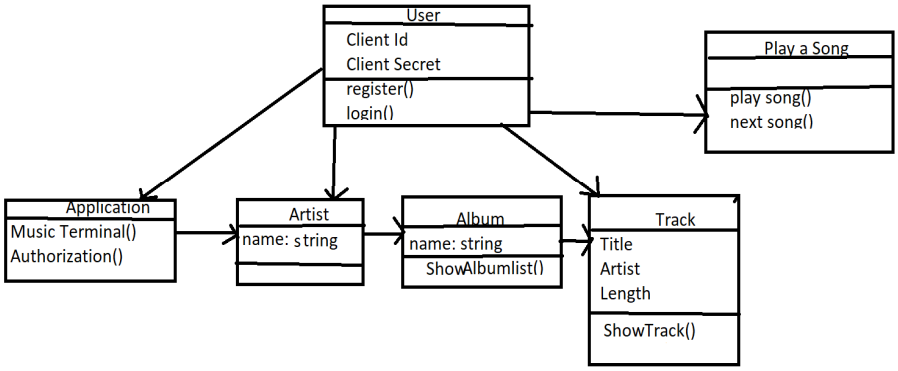
1) *Data Abstraction:* This is the process of describing important features without incorporating background information orexplanations.
2) *Elaboration is the Process of Refinement:* A hierarchy is created by step-by-step dissecting a macro-statement of function until programming language statements are reached. Abstraction and refinement are two notions that work together.
3) *Modularity:* Software architecture is separated into modules, which are individual components.
4) *Software Architecture:* This term refers to the general structure of software as well as the methods in which that structuremaintains a system's conceptual integrity.
5) *Control Hierarchy:* A programme structure that denotes a control hierarchy and describes the organisation of a programme component.
6) *Data Structure:* A data structure is a representation of the logical relationship between distinct data pieces.
7) *Software Procedure:* It focuses on each module's processing separately.
8) *Information Hiding:* Modules should be described and constructed in such a way that information contained inside them is unavailable to other modules that don't require it.

VII. UML DIAGRAMS

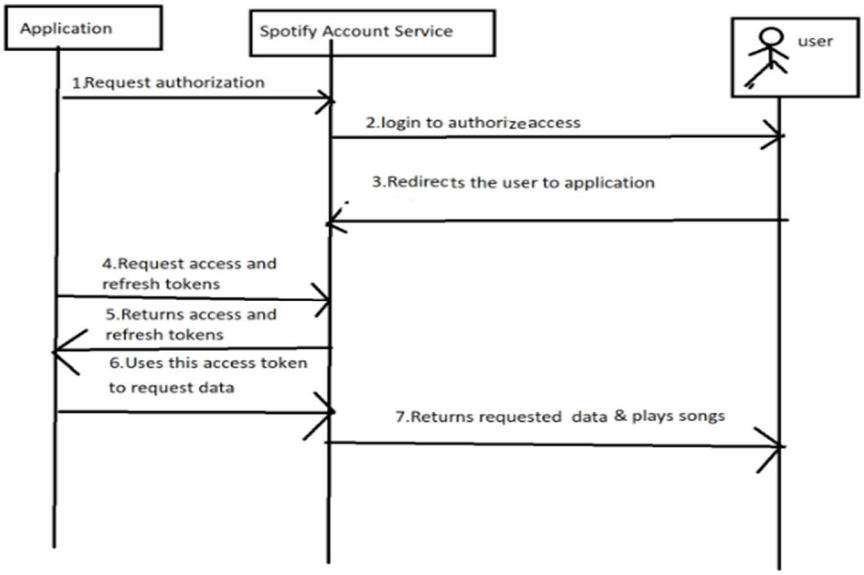
A. *Use-Case Diagram*



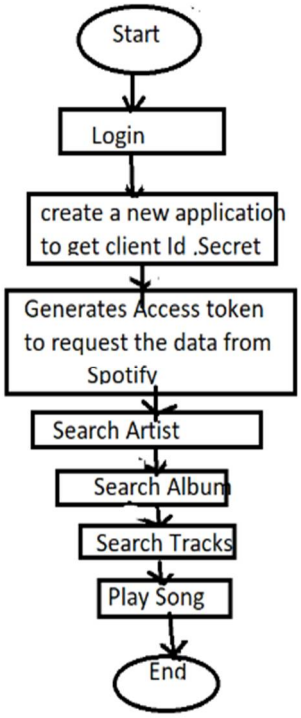
B. *Class Diagram*



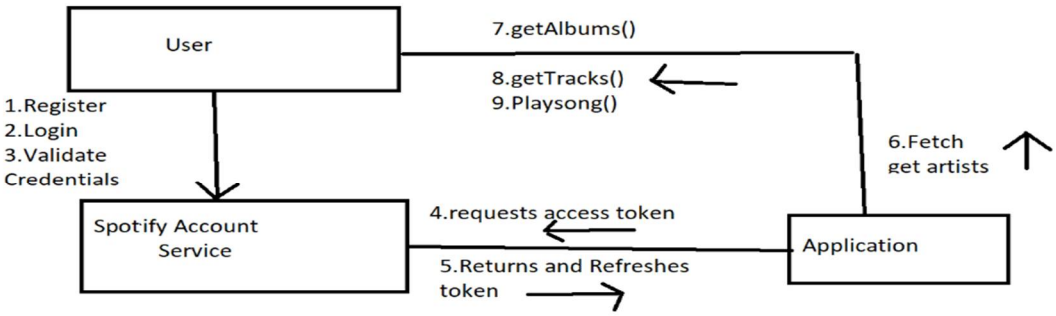
C. Sequence Diagram



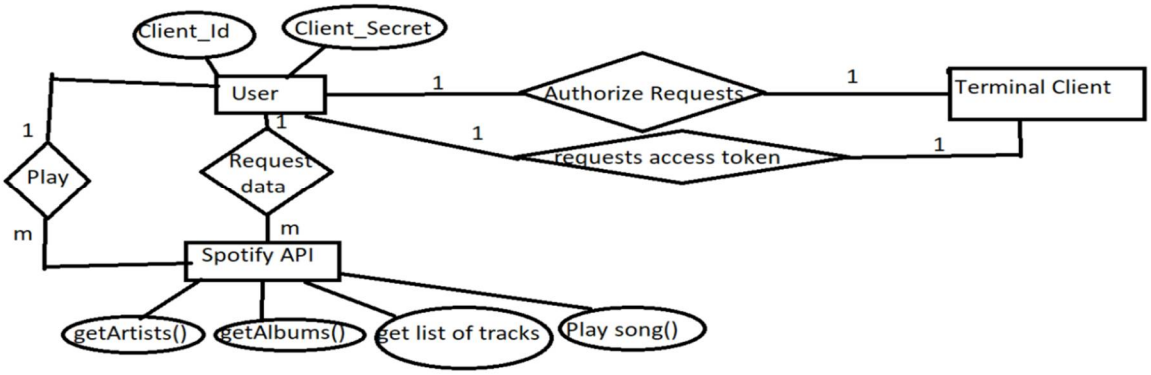
D. Activity Diagram



E. Collaboration Diagram

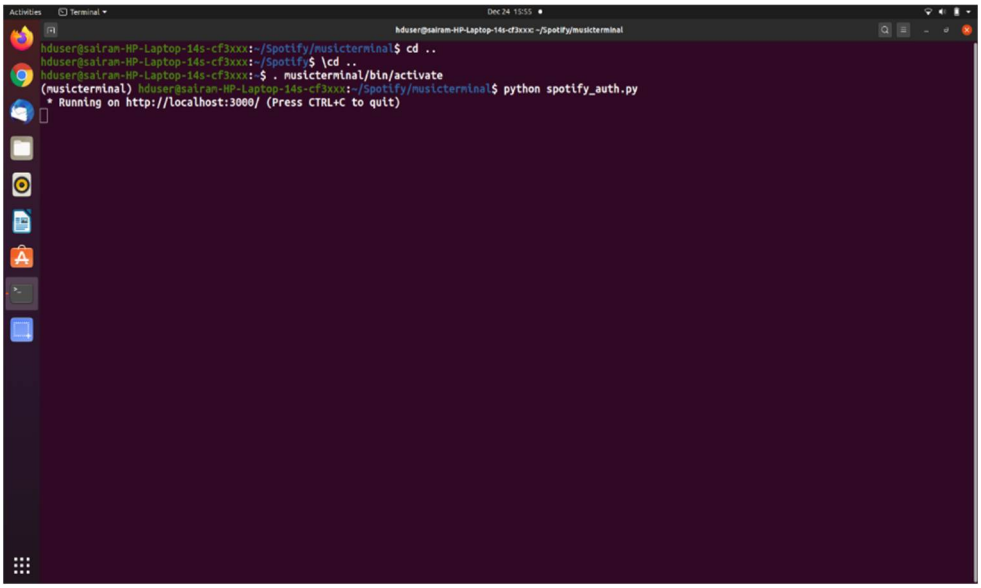


F. Entity-Relationship Diagram

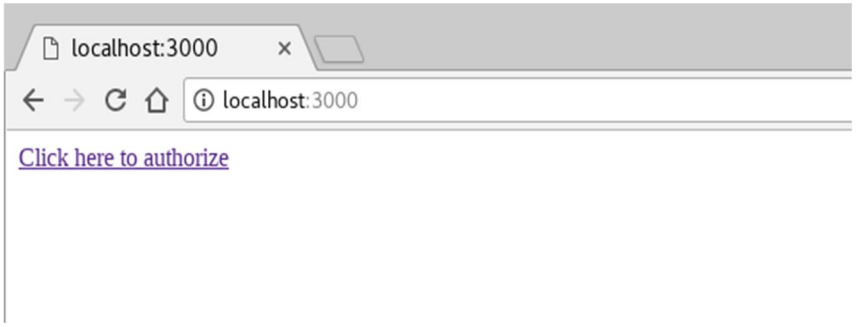


VIII. RESULTS

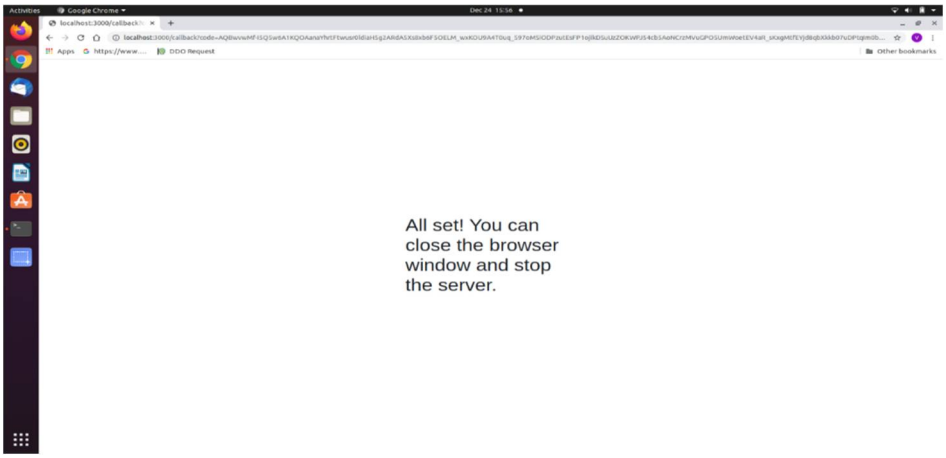
A. Terminal Client



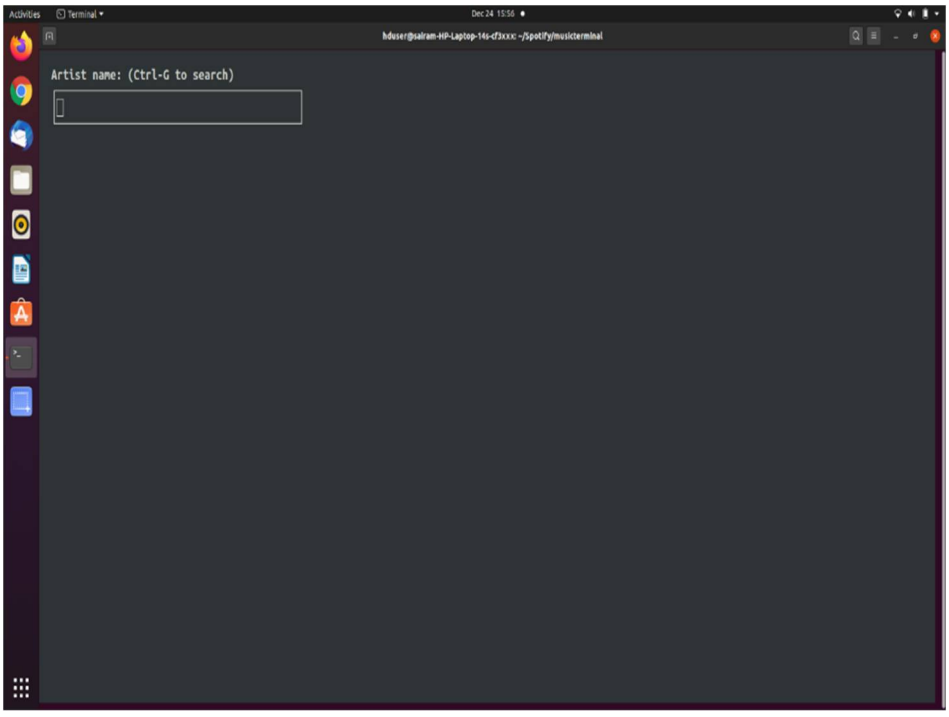
B. Running on http://localhost:3000/ (Press CTRL+C to quit)

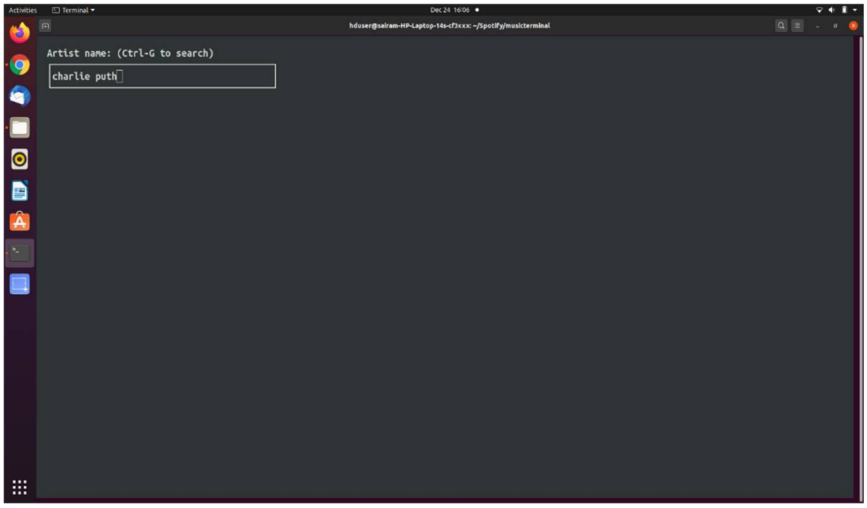


C. Authorized Page

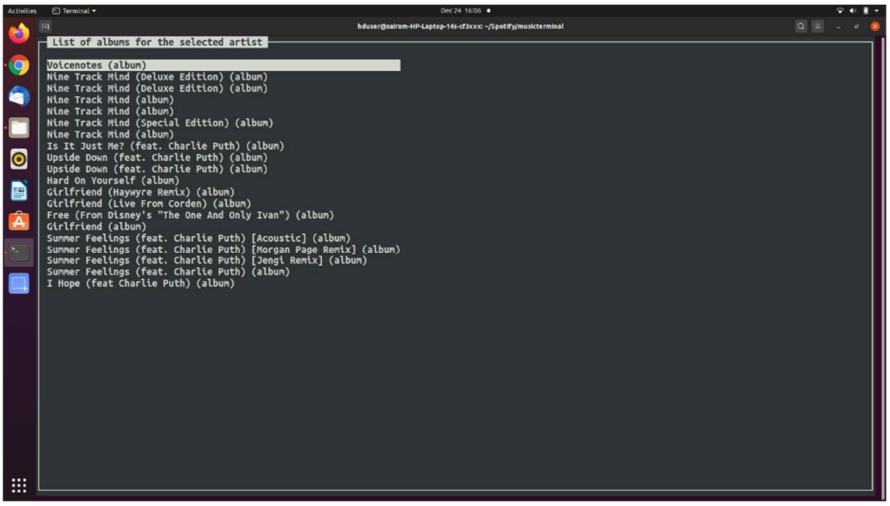


D. Search Artist

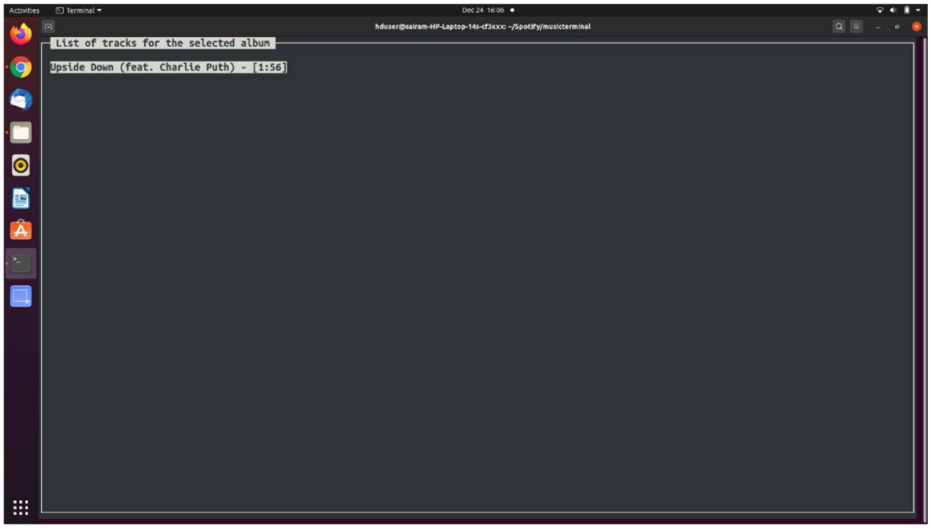




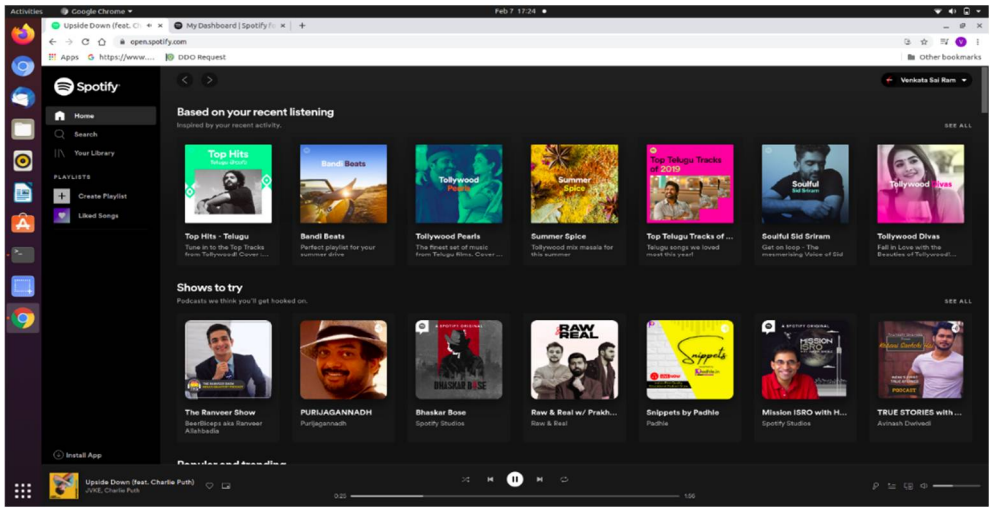
E. Browsing Artist's Albums



F. Browsing list of tracks



G. Playing a Song



IX. CONCLUSION

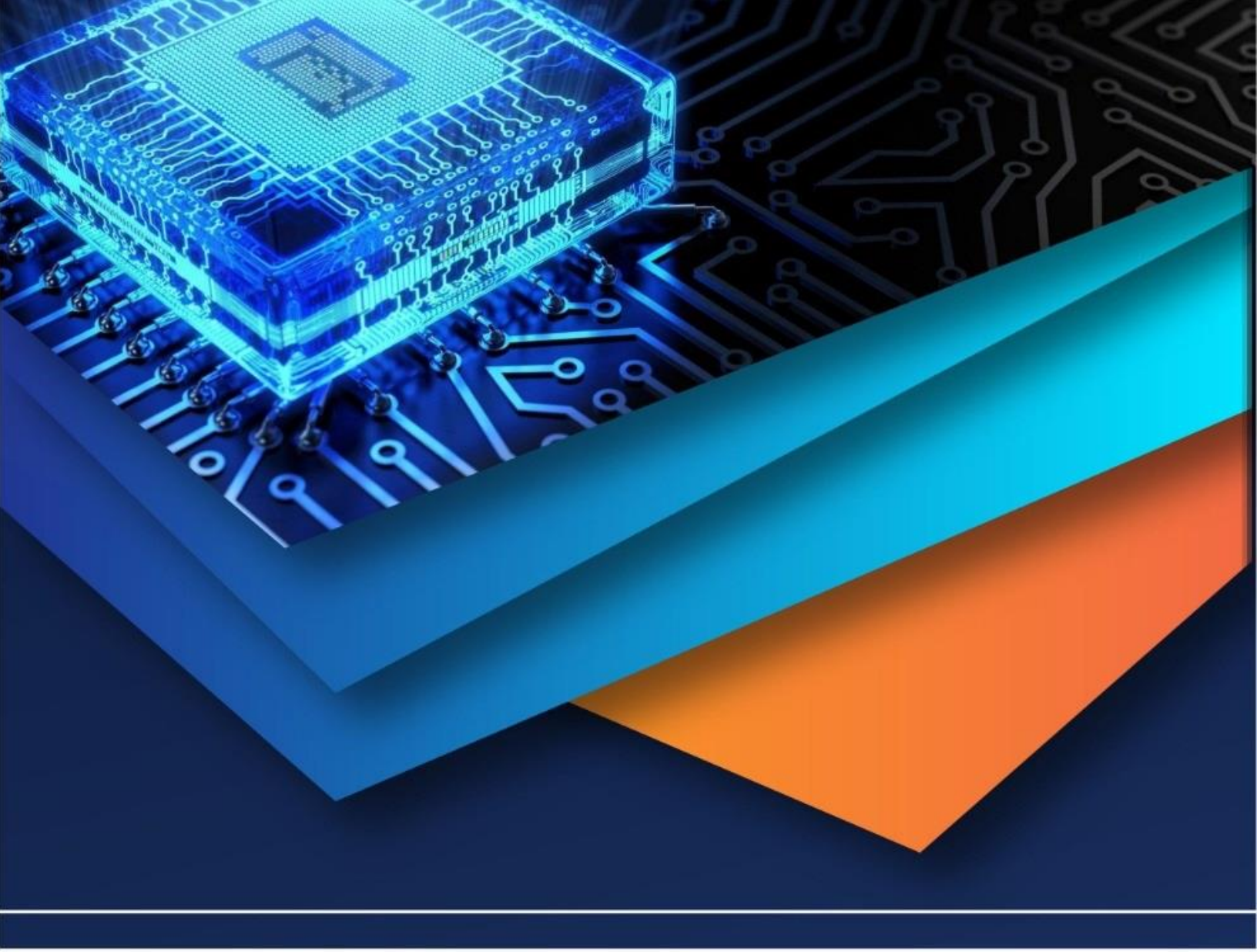
We started by creating an application on Spotify and learning our way around its developer's website. Then, we learned how to implement the two types of authentication flow that Spotify supports: the client credentials flow and the authorization flow. We also implemented a whole module wrapper with some of the functionality available from Spotify's REST API. Then, we implemented a simple terminal client where users can search for artists, browse the artist's albums and tracks, and finally play a song in the user's active device, which can be a computer, mobile phone, or even a video game console.

X. FUTURE SCOPE

The Spotify Web API endpoints return JSON metadata about music artists, albums, and tracks, directly from the Spotify Data Catalogue. We can improve our modules by providing access to user related data, like playlists and music that the user saves in the Your Music library using Web API.

REFERENCES

[1] "Spotify Music". Roku Channel Store. Roku. Retrieved 12 November 2018.
[2] Bertoni, Steven. "Spotify's Daniel Ek: The Most Important Man In Music". Forbes. Retrieved 31 May 2020.
[3] "What is Spotify", How it Works Book of Amazing Technology, Imagine Publishing, p. 113, 2011, ISBN 978-1-908222-0-84
[4] Clark, Bryan . "Spotify Codes bring Snapchat-like QR codes to music streaming". The Next Web. Retrieved 6 May 2017
[5] Geere, Duncan. "Spotify now top-tier music revenue source in Sweden". Wired. Archived from the original on 30 October 2010. Retrieved 22 January 2017.
[6] "Press: Background information". Spotify. Archived from the original on 2012-03-25. Retrieved 2012-03-27.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY