Spotify for Developers

DeveloperDeveloperDeveloper!
UK Community Events

Developer Day 2019 #DDD14

Spotify®

Music Services for Developers

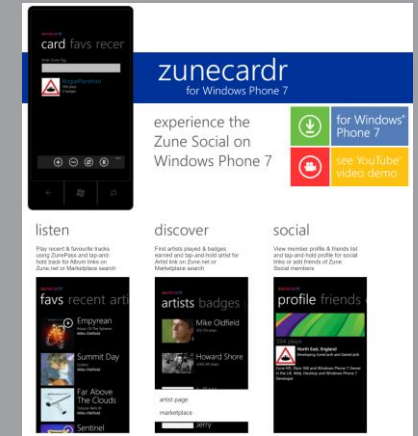Spotify for Developers
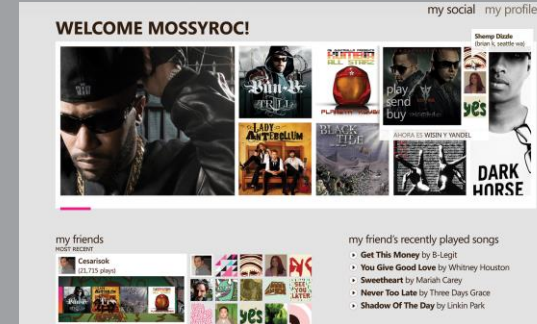
Spotify®

Zune for Developers

Microsoft Music Hardware, Software and Marketplace Ecosystem

Zune Social XML from flash-based Zune Cards and undocumented API

ZuneCardr - Zune Social on Windows Phone 7 with over 330,000 downloads

Music Services for Developers
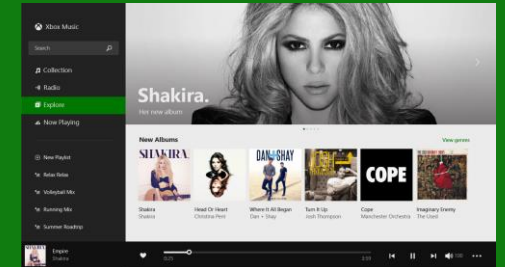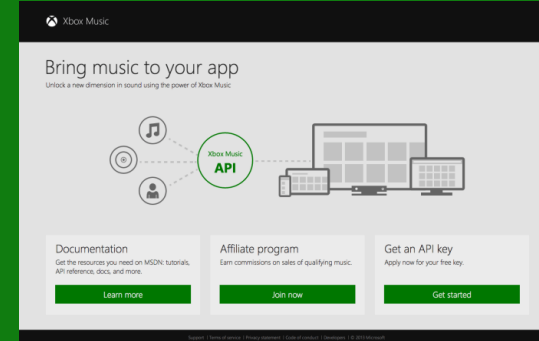
Xbox Music for Developers

Microsoft music service - rebranded from Zune Music Service

Fully-documented API, Developer Pilot programme

AI-enhanced image service & open-source SDK
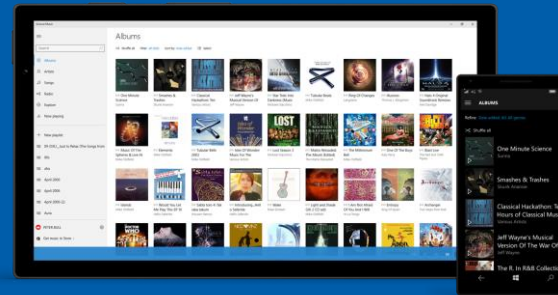
Music Services for Developers

Xbox Music

Groove for Developers

Microsoft music service, rebranded from Xbox Music – ended 2018

Fully-documented API, full music playback and open-source SDK

TuneCardr – brought back core ZuneCardr features to Windows 10

Music Services for Developers
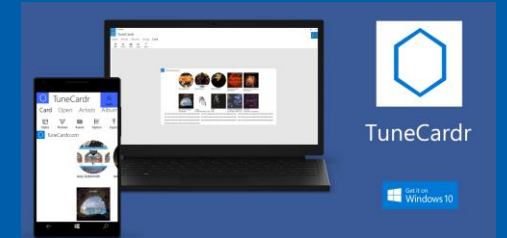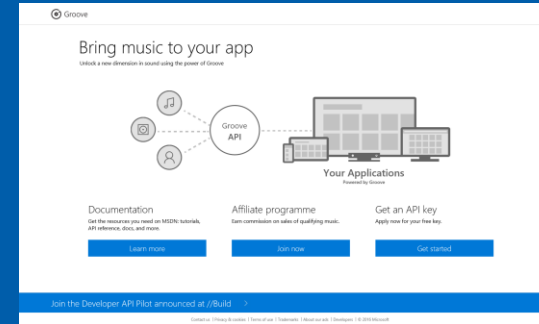
Groove

Spotify for Developers

Independent music service, ex-Groove users optionally transitioned to Spotify

Fully-documented API & interactive web console to test API calls online

First-party JavaScript, iOS & Android SDKs and third-party SDKs on NuGet

Music Services for Developers

Spotify®

developer.spotify.com

Spotify for Developers

Spotify®

Dashboard
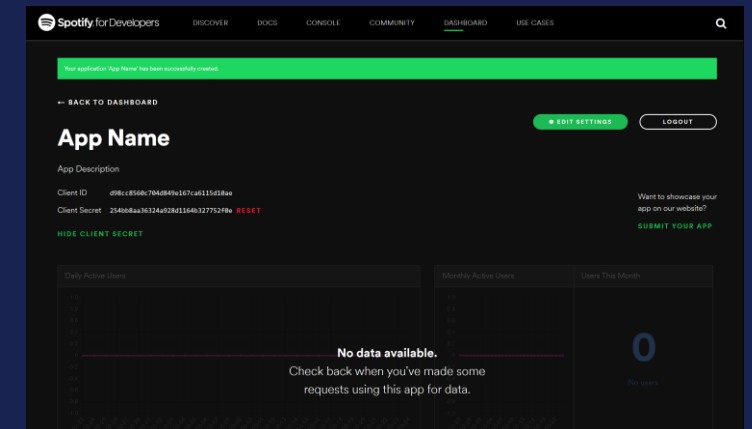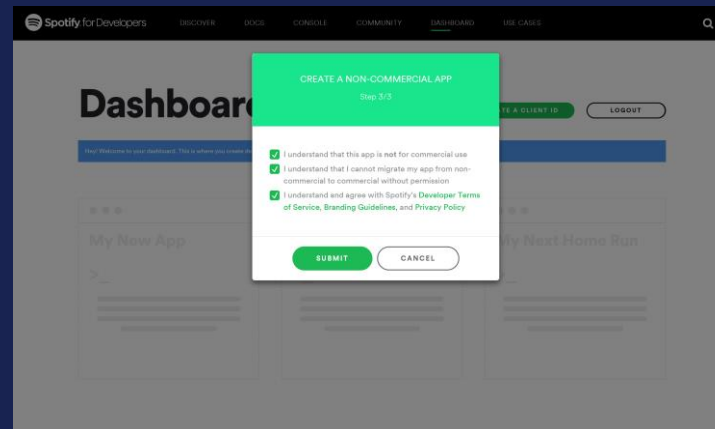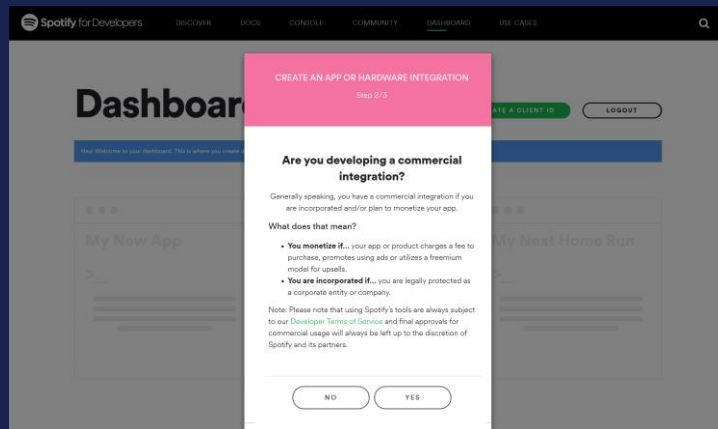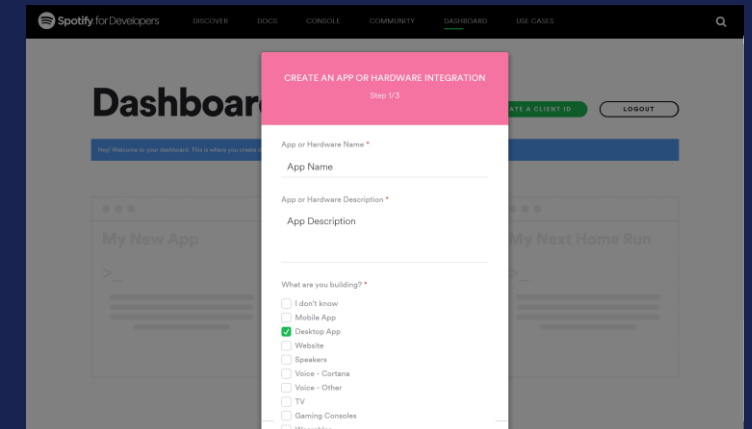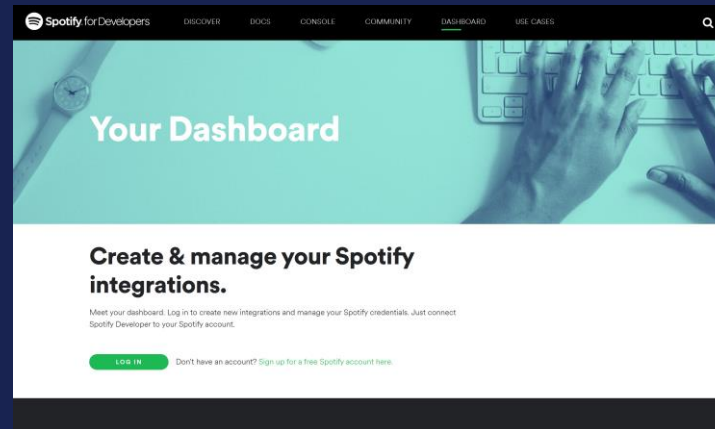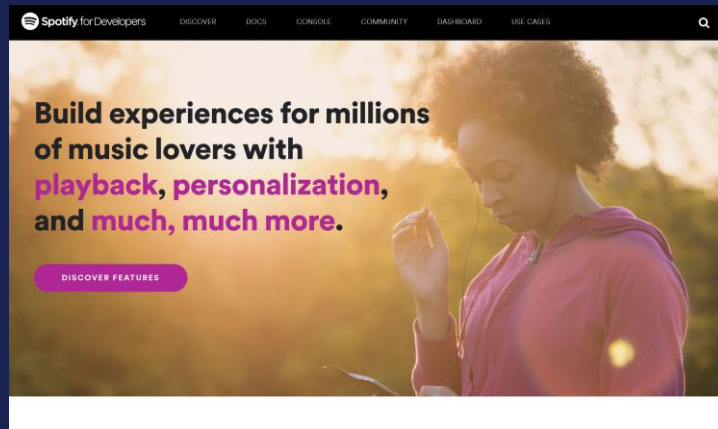
developer.spotify.com

Authorisation Guide

Spotify for Developers

USER · APPLICATION · SPOTIFY
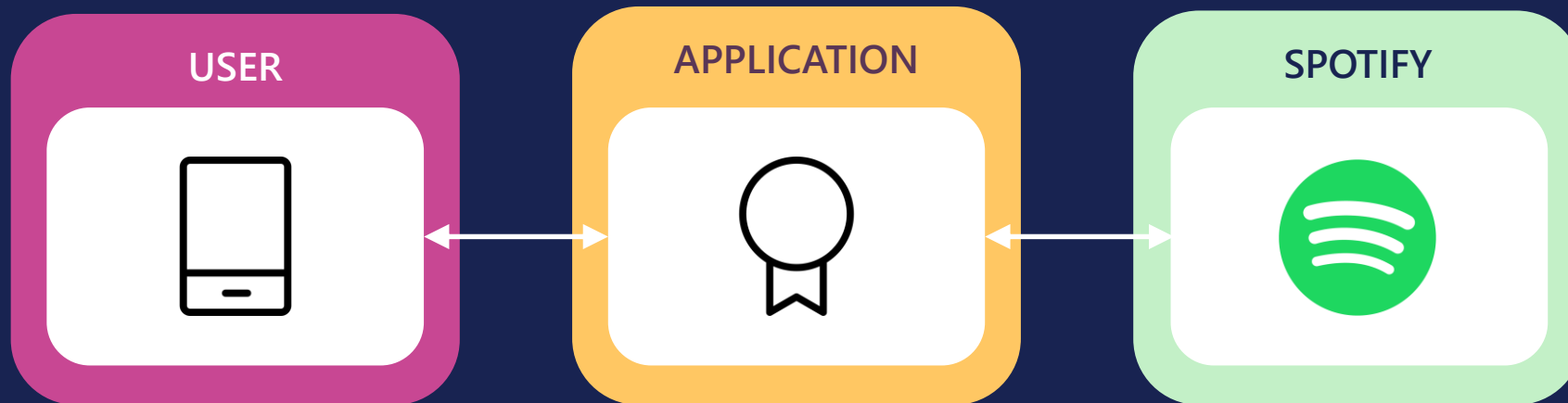
Authorisation Guide

Authorisation Scopes

Listening History

user-read-recently-played

user-top-read

Library

user-library-modify

user-library-read

Playlists

playlist-read-private

playlist-modify-public

playlist-modify-private

playlist-read-collaborative

Users

user-read-email

user-read-birthdate

user-read-private

ugc-image-upload*

Spotify Connect

user-read-playback-state

user-modify-playback-state

user-read-currently-playing

Playback

app-remote-control

streaming

Follow

user-follow-read

user-follow-modify

Authorisation Code Flow

Authorisation Guide

Spotify®

APPLICATION

Request authorisation to access data

client_id
response_type
redirect_url
state
scope

ACCOUNTS SERVICE

Display scopes & prompt user to login (if required)

USER

Log In Authorise Access

code
state

Request Access & Refresh Tokens

client_id
client_secret
grant_type
code
redirect_url

Return Access & Refresh Tokens

Access Token Requests to Web API

client_id
response_type
redirect_url
state
scope

SPOTIFY WEB API

Return requested Data

User/Access Token Requests to Web API

access_token

JSON Object

Return new Access Token

access_token

Authorisation Code Flow

Access Code Request

Query Parameter	
client_id	Client ID provided to you by Spotify when you registered your application
response_type	Set to "code"
redirect_uri	URI to redirect to after the user grants/denies permission. This URI needs to be entered in the URI whitelist that you specify when you registered your application
state	Strongly recommended as state can be useful for correlating requests and responses
scope	A space-separated list of scopes
show_dialog	Whether or not to force user to approve the app again if they've already done so

Example
https://accounts.spotify.com/authorize?client_id=5fe01282e44241328a84e7c5cc169165&response_type=code&redirect_uri=https%3A%2F%2Fexample.com%2Fcallback&scope=user-read-private%20user-read-email&state=DDD

Authorisation Code Flow

Access Code Response

User Accepts Request

https://example.com/callback?code=NApCCg..BkWtQ&state=DDD	
code	Authorisation code that can be exchanged for Token
state	Value of the state parameter supplied in the request

User Denies Request or Error Occurred

https://example.com/callback?error=access_denied&state=DDD	
error	Reason authorisation failed, for example: "access_denied"
state	Value of the state parameter supplied in the request

Authorisation Code Flow

Authorisation Code Request

Header	
Authorization	Base 64 encoded string containing client ID and client secret

Body Parameter	
grant_type	Set to "authorization_code"
code	Authorisation code returned from Access Code Request
redirect_uri	URI to redirect to after the user grants/denies permission. This URI needs to be entered in the URI whitelist that you specify when you registered your application

Authorisation Code Flow

Authorisation Code Response

POST https://accounts.spotify.com/api/token	
Header	**Body**

Authorization: Basic *<base64 encoded client_id:client_secret>*	{
	  "access_token": "NgCXRK...MzYjw",
	  "token_type": "Bearer",
	  "scope": "user-read-private user-read-email",
	  "expires_in": 3600,
	  "refresh_token": "NgAagA...Um_SHo"
	}

Authorisation Code Flow

Authorisation Code Result

Response	
access_token	Access token that can be provided in subsequent calls e.g. Spotify Web API services
token_type	How the access token may be used - always "Bearer"
scope	Space-separated list of scopes which have been granted for this access_token
expires_in	Time period (in seconds) for which the access token is valid.
refresh_token	Token that can be sent to Spotify Accounts service in place of authorisation code

Authorisation Code Flow

Refresh Token Request

Header	
Authorization	Base 64 encoded string containing client ID and client secret

Body Parameter	
grant_type	Set to "refresh_token"
refresh_token	Refresh Token returned from the Authorisation Code exchange

Authorisation Code Flow

Refresh Token Response

POST https://accounts.spotify.com/api/token	
Header	**Body**

Authorization: Basic *<base64 encoded client_id:client_secret>*	{
	 "access_token": "NgA6ZcYI...ixn8bUQ",
	 "token_type": "Bearer",
	 "scope": "user-read-private user-read-email",
	 "expires_in": 3600
	 }

Authorisation Code Flow

Refresh Token Result

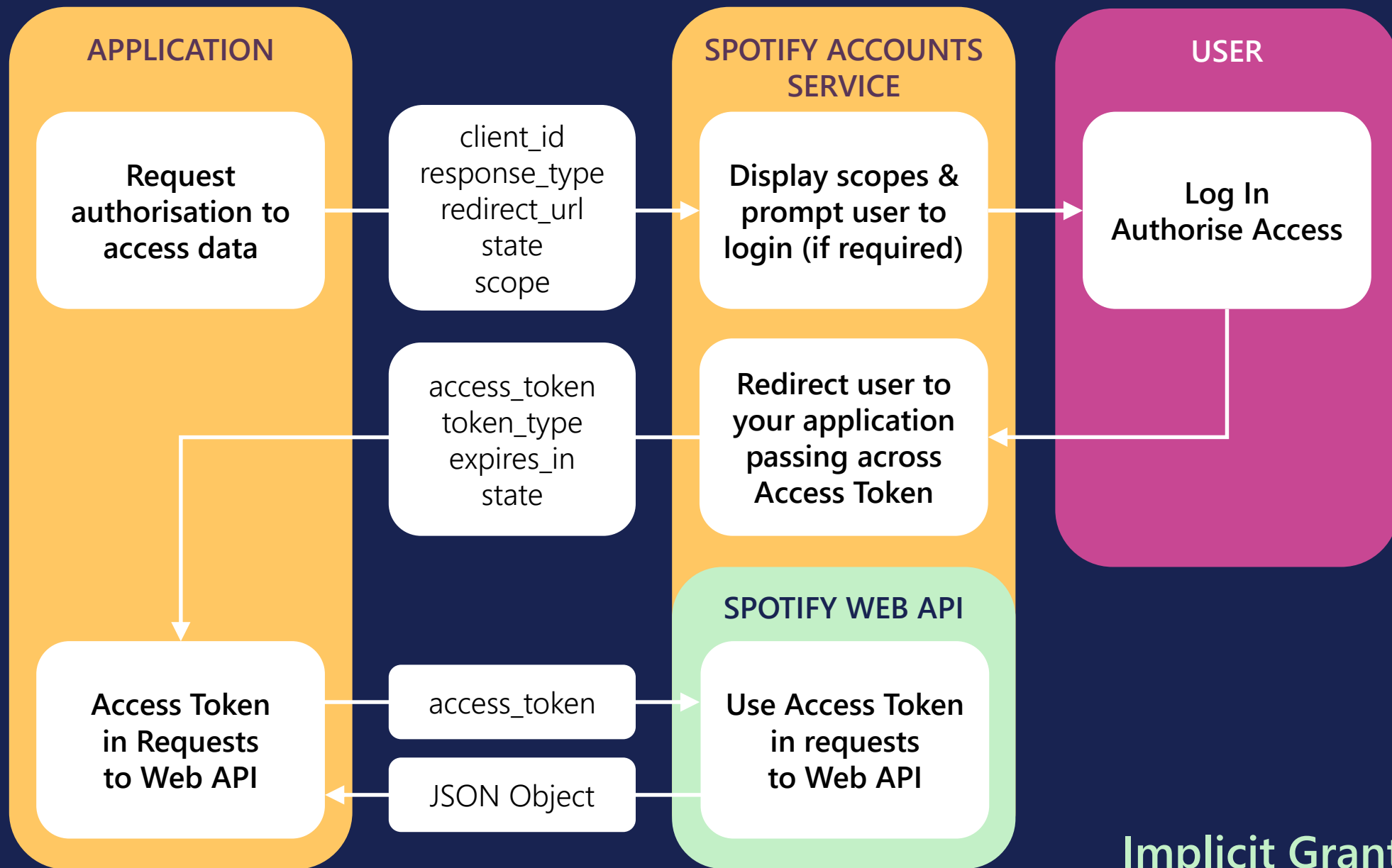| Response	
access_token	Access token that can be provided in subsequent calls e.g. Spotify Web API services
token_type	How the access token may be used - always "Bearer"
scope	Space-separated list of scopes which have been granted for this access_token
expires_in	Time period (in seconds) for which the access token is valid.

Authorisation Code Flow

Implicit Grant Flow

Authorisation Guide

Spotify®

APPLICATION

Request authorisation to access data

client_id
response_type
redirect_url
state
scope

access_token
token_type
expires_in
state

Access Token in Requests to Web API

access_token

JSON Object

SPOTIFY ACCOUNTS SERVICE

Display scopes & prompt user to login (if required)

Redirect user to your application passing across Access Token

USER

Log In
Authorise Access

SPOTIFY WEB API

Use Access Token in requests to Web API

Implicit Grant Flow

Implicit Grant Request

Query Parameter	
client_id	Client ID provided to you by Spotify when you registered your application
response_type	Set to "token"
redirect_uri	URI to redirect to after the user grants/denies permission. This URI needs to be entered in the URI whitelist that you specify when you registered your application
state	Strongly recommended as state can be useful for correlating requests and responses
scope	A space-separated list of scopes
show_dialog	Whether or not to force user to approve the app again if they've already done so

Example
https://accounts.spotify.com/authorize?client_id=5fe01282e94241328a84e7c5cc169164&redirect_uri=http:%2F%2Fexample.com%2Fcallback&scope=user-read-private%20user-read-email&response_type=token&state=DDD

Implicit Grant Flow

Implicit Grant Response

User Grants Access

https://example.com/callback#access_token=Ac...Tk&token_type=Bearer&expires_in=3600&state=DDD	
access_token	Token that can be provided in subsequent calls
token_type	Set to "Bearer"
expires_in	The time period (in seconds) for which the access token is valid
state	The value of the state parameter supplied in the request

User Denies Request or Error Occurred
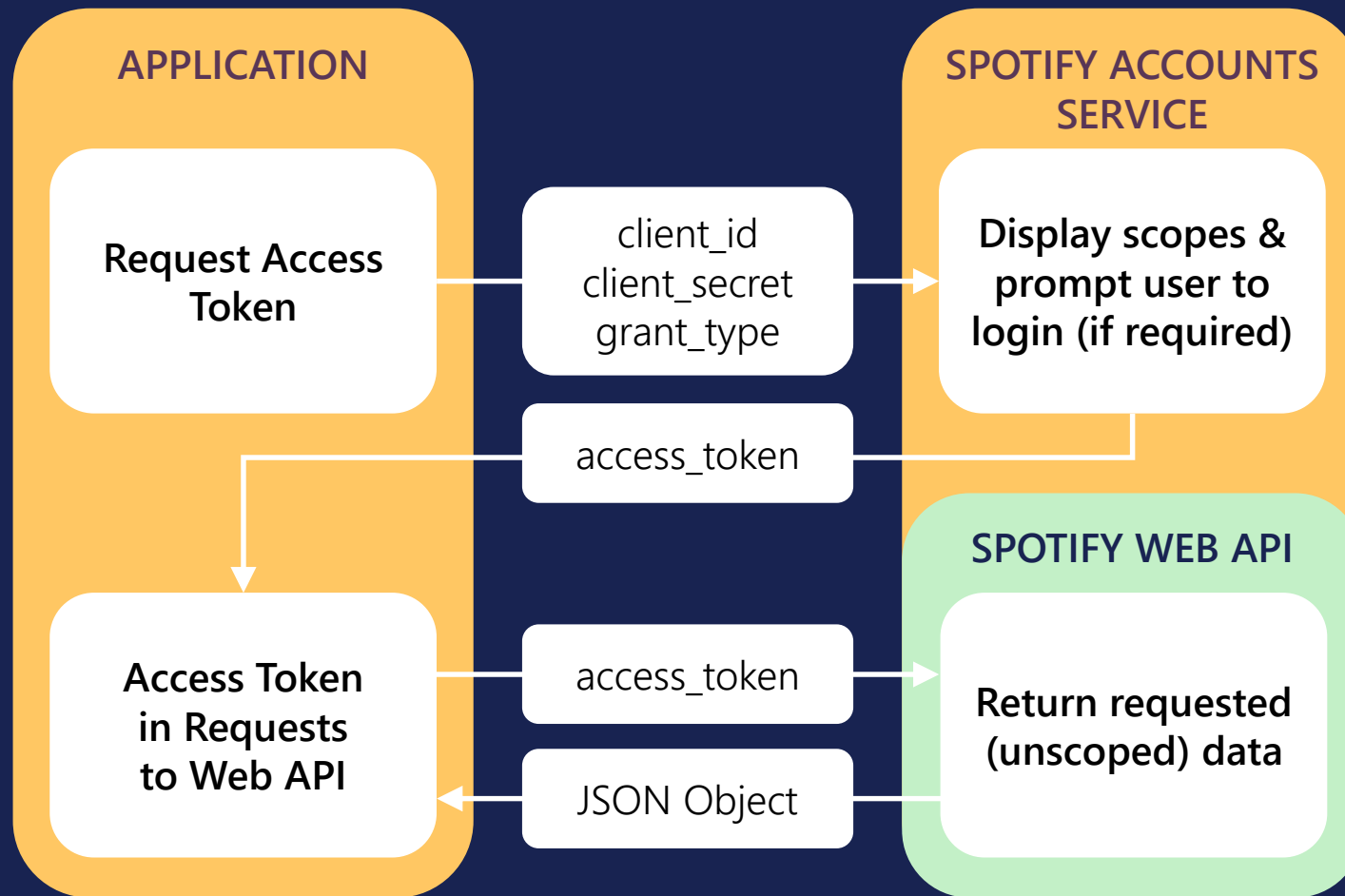
https://example.com/callback?error=access_denied&state=DDD	
error	Reason authorisation failed, for example: "access_denied"
state	Value of the state parameter supplied in the request

Implicit Grant Flow

Client Credentials Flow

Authorisation Guide

Spotify®

Client Credentials Request

Header	
Authorization	Base 64 encoded string that contains client ID and client secret key Format: Authorization: Basic <base64 encoded client_id:client_secret>

Body Parameter	
grant_type	Set to "client_credentials"

Client Credentials Flow

Client Credentials Response

POST https://accounts.spotify.com/api/token	
Header	**Response**

Authorization: Basic *<base64 encoded client_id:client_secret>*	{
	  "access_token": "NgCXRKc...MzYjw",
	  "token_type": "bearer",
	  "expires_in": 3600,
	}

Client Credentials Flow

Client Credentials Result

Response	
access_token	Access token that can be provided in subsequent calls e.g. Spotify Web API services
token_type	How the access token may be used - always "bearer"
expires_in	Time period (in seconds) for which the access token is valid.

Client Credentials Flow

Authorisation Flows

Demo

Spotify®

Authorisation Code Flow - Postman   **Demo**

Implicit Grant Flow - Postman   Demo

```csharp
using System;
using System.Collections.Generic;
using System.Net.Http;
using System.Net.Http.Headers;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;

namespace Spotify.Demo.Authentication
{
    0 references
    class Program
    {
        0 references
        static async Task Main(string[] args)
        {
            var token = string.Empty;
            using (var auth = new HttpClient())
            {
                auth.DefaultRequestHeaders.Authorization =
                new AuthenticationHeaderValue("Basic", Convert.ToBase64String(Encoding.ASCII.GetBytes(
                "6a53fb50cb7949709041bdc74c75c044:b9e7de1be14946e0ad080c0c417266aa")));
                var response = await auth.PostAsync(new Uri("https://accounts.spotify.com/api/token"),
                new FormUrlEncodedContent(new Dictionary<string, string>()
                { { "grant_type", "client_credentials" } }));
                token = Regex.Match(await response.Content.ReadAsStringAsync(),
                ".*\"access_token\":\"(.*?)\".*", RegexOptions.IgnoreCase).Groups[1].Value;
            }
            Console.WriteLine(token);
            Console.ReadLine();
        }
    }
}
```
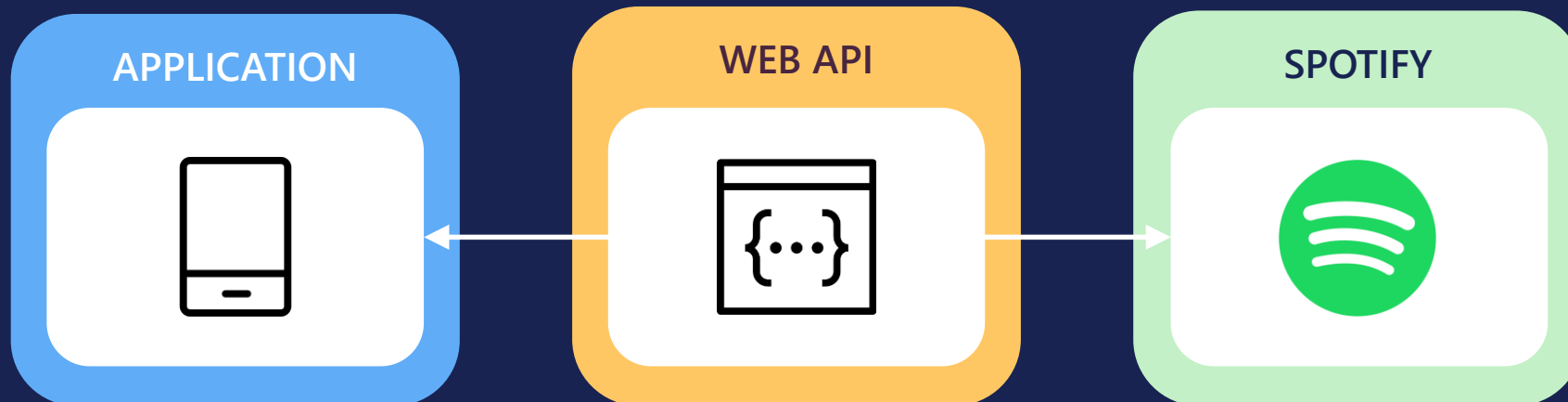
Client Credentials – Console App (.NET Core)   Demo

APPLICATION — WEB API — SPOTIFY

Spotify Web API

Spotify Web API Requests

Method	Action
GET	Retrieves Resources
POST	Creates Resources
PUT	Changes and/or replaces Resources or Collections
DELETE	Deletes Resources

Spotify Web API

Spotify Web API Responses

Objects Index
AlbumObject
ArtistObject
AudioFeaturesObject
CategoryObject
ContextObject
CurrentlyPlayingObject
CursorObject
CursorPagingObject
DeviceObject

Objects Index
DevicesObject
ErrorObject
ExternalIdObject
PagingObject
PlayHistoryObject
PlaylistObject
PlaylistTrackObject
PrivateUserObject
PublicUserObject

Objects Index
RecommendationSeedObject
RecommendationsResponseObject
SavedAlbumObject
SavedTrackObject
SimplifiedAlbumObject
SimplifiedTrackObject
TrackObject
TuneableTrackObject

Spotify Web API

Spotify Web API App Authorisation

Spotify®

App Authorisation

Search

Search for an Item

Browse

Get All Categories

Get a Category

Get a Category's Playlists

Get Recommendations

Get Recommendation Genres

Get All New Releases

Get All Featured Playlists

Playlists

Get a Playlist

Get a Playlist's Tracks

Artists

Get Multiple Artists

Get an Artist

Get an Artist's Albums

Get an Artist's Top Tracks

Get an Artist's Related Artists

Albums

Get Multiple Albums

Get an Album

Get an Album's Tracks

Tracks

Get Audio Features for a Track

Get a Track

Get Audio Analysis for a Track

Get Audio Features for Several Tracks

Get Several Tracks

Spotify Web API App Authorisation

Get All
New Releases

Spotify Web API App Authorisation
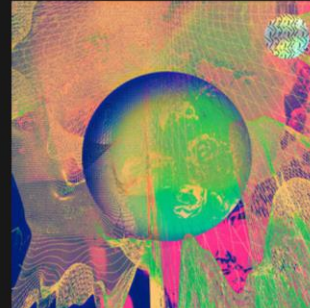
Spotify®

Get All New Releases

Request

Header	
Authorization	Valid Access Token from Spotify Accounts service

Query Parameter	
country	ISO 3166-1 alpha-2 country code e.g. "GB"
limit	Maximum number of results to return
offset	Index of first result to return

Get All New Releases

```
{
  "albums" : {
    "href" : "https://api.spotify.com/v1/browse/new-releases?country=GB&offset=0&limit=20",
    "items" : [ {
      "album_type" : "single",
      "artists" : [ {
        ...
        "href" : "https://api.spotify.com/v1/artists/2RdwBSPQiwcmiDo9kixcl8",
        "id" : "2RdwBSPQiwcmiDo9kixcl8",
        "name" : "Pharrell Williams",
        "type" : "artist",
        "uri" : "spotify:artist:2RdwBSPQiwcmiDo9kixcl8"
      } ],
      ...
      "href" : "https://api.spotify.com/v1/albums/5ZX4m5aVSmWQ5iHAPQpT71",
      "id" : "5ZX4m5aVSmWQ5iHAPQpT71",
      "images" : [ {
        "height" : 640,
        "url" : "https://i.scdn.co/image/e6b635ebe3ef4ba22492f5698a7b5d417f78b88a",
        "width" : 640
      }],
      "name" : "Runnin'",
      "type" : "album",
      "uri" : "spotify:album:5ZX4m5aVSmWQ5iHAPQpT71"
    }, {
      "album_type" : "single",

  "artists" : [ {
        ...
        "href" : "https://api.spotify.com/v1/artists/3TVXtAsR1lnumwj472S9r4",
        "id" : "3TVXtAsR1lnumwj472S9r4",
        "name" : "Drake",
        "type" : "artist",
        "uri" : "spotify:artist:3TVXtAsR1lnumwj472S9r4"
      } ],
      ...
      "href" : "https://api.spotify.com/v1/albums/0geTzdk2lnlqloB16fW9Nd",
      "id" : "0geTzdk2lnlqloB16fW9Nd",
      "images" : [ {
        "height" : 640,
        "url" : "https://i.scdn.co/image/d40e9c3d22bde2fbdb2ecc03cccd7a0e77f42e4c",
        "width" : 640
      }],
      "name" : "Sneakin'",
      "type" : "album",
      "uri" : "spotify:album:0geTzdk2lnlqloB16fW9Nd"
    }, {
      ...
    } ],
    "limit" : 20,
    "next" : "https://api.spotify.com/v1/browse/new-releases?country=GB&offset=20&limit=20",
    "offset" : 0,
    "previous" : null,
    "total" : 500
  }
}
```

Get All New Releases   Response

Get All New Releases Spotify Web API

Demo

Spotify®

Get All New Releases – Postman   **Demo**

```csharp
public class IndexModel : PageModel
{
    private readonly ILogger<IndexModel> _logger;

    0 references
    public IndexModel(ILogger<IndexModel> logger)
    {
        _logger = logger;
    }

    2 references
    public dynamic NewReleases { get; set; }

    0 references
    public async Task OnGetAsync()
    {
        var token = "BQADjLEZUQR9NMCH6BdTMVrzHx565SQSxg66zHT3S2pmceIS-A5XxU5p1Zz
        using (var client = new HttpClient())
        {
            client.DefaultRequestHeaders.Authorization =
            new AuthenticationHeaderValue("Bearer", token);
            var response = await client.GetAsync(new Uri(
                "https://api.spotify.com/v1/browse/new-releases?country=GB"
            ));
            NewReleases = await response.Content.ReadAsAsync<dynamic>();
        }
    }
}
```

Get All New Releases – ASP.NET Core Web Application   **Demo**

User Authorisation ...

Follow	Playlists	Library
Get Following State for Artists/Users	Add Tracks to a Playlist	Check User's Saved Albums
Check if Users Follow a Playlist	Remove Tracks from a Playlist	Save Tracks for User
Follow Artists or Users	Get a Playlist Cover Image	Remove Albums for Current User
Follow a Playlist	Upload Custom Playlist Cover Image	Save Albums for Current User
Get User's Followed Artists	Get a List of Current User's Playlists	Remove User's Saved Tracks
Unfollow Artists or Users	Change a Playlist's Details	Get User's Saved Albums
Unfollow Playlist	Get a List of a User's Playlists	Get User's Saved Tracks
	Replace a Playlist's Tracks	Check User's Saved Tracks
	Create a Playlist	
	Reorder a Playlist's Tracks	

Spotify Web API User Authorisation

... User Authorisation

Player
Skip User's Playback To Next Track
Seek to Position In Playing Track
Get a User's Available Devices
Toggle Shuffle For User's Playback
Transfer a User's Playback
Get User's Recently Played Tracks
Start/Resume a User's Playback

Player
Set Repeat Mode On User's Playback
Pause a User's Playback
Skip Playback To Previous Track
Get Information About Playback
Get User's Currently Playing Track
Set Volume For User's Playback

Personalisation
Get a User's Top Artists and Tracks

User Profile
Get a User's Profile
Get Current User's Profile

Spotify Web API User Authorisation

Get User's Followed Artists

Spotify Web API User Authorisation

Spotify®

Get User's Followed Artists

Request

Header & Scope	
Authorization	User Token from Spotify Accounts service **user-follow-read**

Query Parameter	
type	ID type, currently only "artist" supported
limit	Maximum number of items to return
after	Last artist ID retrieved from previous request

Get User's Followed Artists

GET https://api.spotify.com/v1/me/following?type=artist

```
{
  "artists" : {
    "items" : [ {
      ...
      "followers" : {
        "href" : null,
        "total" : 7753
      },
      "genres" : [ "swedish hip hop" ],
      "href" : "https://api.spotify.com/v1/artists/0l2XqVXqHScXjHhk6AYYRe",
      "id" : "0l2XqVXqHScXjHhk6AYYRe",
      "images" : [ {
        "height" : 640,
        "url" : "https://i.scdn.co/image/2c8c0cea05bf3d3c070b7498d8d0b957c4cdec20",
        "width" : 640
      }],
      "name" : "Afasi & Filthy",
      "popularity" : 54,
      "type" : "artist",
      "uri" : "spotify:artist:0l2XqVXqHScXjHhk6AYYRe"
    },{
      ...
    }],
```

```
    "artists" : [ {
        ...
        "href" : "https://api.spotify.com/v1/artists/3TVXtAsR1Inumwj472S9r4",
        "id" : "3TVXtAsR1Inumwj472S9r4",
        "name" : "Drake",
        "type" : "artist",
        "uri" : "spotify:artist:3TVXtAsR1Inumwj472S9r4"
      } ],
      ...
      "href" : "https://api.spotify.com/v1/albums/0geTzdk2InlqIoB16fW9Nd",
      "id" : "0geTzdk2InlqIoB16fW9Nd",
      "images" : [ {
        "height" : 640,
        "url" : "https://i.scdn.co/image/d40e9c3d22bde2fbdb2ecc03cccd7a0e77f42e4c",
        "width" : 640
      }],
      "name" : "Sneakin'",
      "type" : "album",
      "uri" : "spotify:album:0geTzdk2InlqIoB16fW9Nd"
    }, {
      ...
    } ],
    "limit" : 20,
    "next" : "https://api.spotify.com/v1/browse/new-releases?country=GB&offset=20&limit=20",
    "offset" : 0,
    "previous" : null,
    "total" : 500
  }
}
```

Get User's Followed Artists   Response

Get User's Followed Artists – Postman   Demo

```csharp
public class IndexModel : PageModel
{
    private readonly ILogger<IndexModel> _logger;

    0 references
    public IndexModel(ILogger<IndexModel> logger)
    {
        _logger = logger;
    }

    2 references
    public dynamic FollowedArtists { get; set; }

    0 references
    public async Task OnGetAsync()
    {
        var token = "BQCL4xUhngJAvpeZWi07oas8DP3YcG3Y1ejk8Z_VNEcOvH5Z9tPCL_tK8lP
        using (var client = new HttpClient())
        {
            client.DefaultRequestHeaders.Authorization =
            new AuthenticationHeaderValue("Bearer", token);
            var response = await client.GetAsync(new Uri(
                "https://api.spotify.com/v1/me/following?type=artist"
            ));
            FollowedArtists = await response.Content.ReadAsAsync<dynamic>();
        }
    }
}
```

Get User's Followed Artists – ASP.NET Core Web Application   Demo

Spotify SDKs

Spotify for Developers

Spotify®

WEB PLAYBACK

iOS

ANDROID

THIRD-PARTY

Spotify SDKs

WEB PLAYBACK

iOS

ANDROID

THIRD-PARTY

Spotify SDKs

WEB PLAYBACK

iOS

ANDROID

THIRD-PARTY

Spotify SDKs

WEB PLAYBACK

iOS
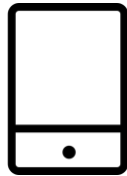
ANDROID

THIRD-PARTY
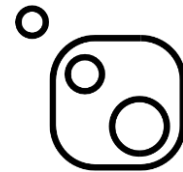
Spotify SDKs

Web Playback SDK

Spotify.Player
#connect
#disconnect
#addListener
#removeListener
#getCurrentState
#setName
#getVolume
#setVolume
#pause
#resume

Spotify.Player
#togglePlay
#seek
#previousTrack
#nextTrack

Events
ready
not_ready
player_state_changed

Error Reference
initialization_error
authentication_error
account_error
playback_error

Objects
WebPlaybackPlayer
WebPlaybackState
WebPlaybackTrack
WebPlaybackError

Spotify Web Playback SDK

```html
<!DOCTYPE html>
<html>
<head>
  <title>Spotify Web Playback SDK Quick Start Tutorial</title>
</head>
<body>
  <h1>Spotify Web Playback SDK Quick Start Tutorial</h1>
  <h2>Open your console log: <code>View > Developer > JavaScript Console</code></h2>

  <script src="https://sdk.scdn.co/spotify-player.js"></script>
  <script>
    window.onSpotifyWebPlaybackSDKReady = () => {
      const token = '[My Spotify Web API access token]';
      const player = new Spotify.Player({
        name: 'Web Playback SDK Quick Start Player',
        getOAuthToken: cb => { cb(token); }
      });

      // Error handling
      player.addListener('initialization_error', ({ message }) => { console.error(message); });
      player.addListener('authentication_error', ({ message }) => { console.error(message); });
      player.addListener('account_error', ({ message }) => { console.error(message); });
      player.addListener('playback_error', ({ message }) => { console.error(message); });

      // Playback status updates
      player.addListener('player_state_changed', state => { console.log(state); });

      // Ready
      player.addListener('ready', ({ device_id }) => {
        console.log('Ready with Device ID', device_id);
      });

      // Not Ready
      player.addListener('not_ready', ({ device_id }) => {
        console.log('Device ID has gone offline', device_id);
      });

      // Connect to the player!
      player.connect();
    };
  </script>
</body>
</html>
```

Spotify Web Playback SDK   **Example**

Spotify.NetStandard

NuGet	nuget.org/packages/Spotify.NetStandard
Package Manager	PM> Install-Package Spotify.NetStandard
.NET CLI	> dotnet add package Spotify.NetStandard

GitHub	github.com/RoguePlanetoid/Spotify-NetStandard
Git	$ git clone https://github.com/RoguePlanetoid/Spotify-NetStandard.git

Example

```
using Spotify.NetStandard.Client;
using Spotify.NetStandard.Requests;

var client = SpotifyClientFactory.CreateSpotifyClient("client-id","client-secret");
var page = new Page() { Limit = 10 };
var browse = await client.LookupNewReleasesAsync(page: page);
foreach (var album in browse.Albums.Items)
{
    Console.WriteLine(album.Name);
}
Console.ReadLine();
```

.NET Standard SDK

ISpotifyClient ...

ISpotifyClient		
AuthAddTracksToPlaylistAsync	AuthLookupFollowedArtistsAsync	AuthLookupUserSavedAlbumsAsync
AuthAsync	AuthLookupFollowingStateAsync	AuthLookupUserSavedTracksAsync
AuthChangePlaylistDetailsAsync	AuthLookupUserFollowingPlaylistAsync	AuthLookupUserTopArtistsAsync
AuthCreatePlaylistAsync	AuthLookupUserPlaybackCurrentAsync	AuthRemoveTracksFromPlaylistAsync
AuthFollowAsync	AuthLookupUserPlaybackCurrentTrackAsync	AuthRemoveUserAlbumsAsync
AuthFollowPlaylistAsync	AuthLookupUserPlaybackDevicesAsync	AuthRemoveUserTracksAsync
AuthGetPlaylistCoverImageAsync	AuthLookupUserPlaylistsAsync	AuthReorderPlaylistTracksAsync
AuthLookupCheckUserSavedAlbumsAsync	AuthLookupUserProfileAsync	AuthReplacePlaylistTracksAsync
AuthLookupCheckUserSavedTracksAsync	AuthLookupUserRecentlyPlayedTracksAsync	AuthSaveUserAlbumsAsync

.NET Standard SDK

... ISpotifyClient

ISpotifyClient		
AuthSaveUserTracksAsync	AuthUserPlaybackPauseAsync	LookupAllCategoriesAsync
AuthUnfollowAsync	AuthUserPlaybackPreviousTrackAsync	LookupArtistAlbumsAsync
AuthUnfollowPlaylistAsync	AuthUserPlaybackSeekTrackAsync	LookupArtistRelatedArtistsAsync
AuthUploadCustomPlaylistImageAsync	AuthUserPlaybackSetRepeatModeAsync	LookupArtistTopTracksAsync
AuthUser	AuthUserPlaybackSetVolumeAsync	LookupAsync
AuthUserAsync	AuthUserPlaybackStartResumeAsync	LookupCategoryAsync
AuthUserImplicit	AuthUserPlaybackToggleShuffleAsync	LookupFeaturedPlaylistsAsync
AuthUserPlaybackNextTrackAsync	AuthUserPlaybackTransferAsync	LookupNewReleasesAsync

.NET Standard SDK

ISpotifyApi ...

ISpotifyApi		
GetAuthorisationCodeAuthUri	GetCategoryPlaylistsAsync	FollowPlaylistAsync
GetAuthorisationCodeAuthTokenAsync	GetRecommendationsAsync	GetUsersFollowedArtistsAsync
GetClientCredentialsAuthTokenAsync	GetRecommendationGenresAsync	UnfollowArtistsOrUsersAsync
GetImplicitGrantAuthUri	GetAllNewReleasesAsync	UnfollowPlaylistAsync
GetImplicitGrantAuthToken	GetAllFeaturedPlaylistsAsync	AddTracksToPlaylistAsync
SearchForItemAsync	GetFollowingStateForArtistsOrUsersAsync	GetPlaylistAsync
GetAllCategoriesAsync	CheckUsersFollowingPlaylistAsync	RemoveTracksFromPlaylistAsync
GetCategoryAsync	FollowArtistsOrUsersAsync	GetPlaylistTracksAsync

.NET Standard SDK

... ISpotifyApi ...

ISpotifyApi		
GetPlaylistCoverImageAsync	CheckUserSavedAlbumsAsync	GetMultipleArtistsAsync
UploadCustomPlaylistCoverImageAsync	SaveUserTracksAsync	GetArtistAsync
GetUserPlaylistsAsync	RemoveUserAlbumsAsync	GetArtistAlbumsAsync
ChangePlaylistDetailsAsync	SaveUserAlbumsAsync	GetArtistTopTracksAsync
GetUserPlaylistsAsync	RemoveUserTracksAsync	GetArtistRelatedArtistsAsync
ReplacePlaylistTracksAsync	GetUserSavedAlbumsAsync	UserPlaybackNextTrackAsync
CreatePlaylistAsync	GetUserSavedTracksAsync	UserPlaybackSeekTrackAsync
ReorderPlaylistTracksAsync	CheckUserSavedTracksAsync	GetUserPlaybackDevicesAsync

.NET Standard SDK

… ISpotifyApi

ISpotifyApi		
UserPlaybackToggleShuffleAsync	GetUserPlaybackCurrentAsync	GetAlbumAsync
UserPlaybackTransferAsync	GetUserPlaybackCurrentTrackAsync	GetAlbumTracksAsync
GetUserRecentlyPlayedTracksAsync	UserPlaybackSetVolumeAsync	GetTrackAudioFeaturesAsync
UserPlaybackStartResumeAsync	GetUserTopArtistsAsync	GetTrackAsync
UserPlaybackSetRepeatModeAsync	GetUserTopTracksAsync	GetTrackAudioAnalysisAsync
UserPlaybackPauseAsync	GetUserProfileAsync	GetSeveralTracksAudioFeaturesAsync
UserPlaybackPreviousTrackAsync	GetMultipleAlbumsAsync	GetSeveralTracksAsync

.NET Standard SDK

Spotify.Uwp

NuGet	nuget.org/packages/Spotify.Uwp
Package Manager	`PM> Install-Package Spotify.Uwp`
.NET CLI	`> dotnet add package Spotify.Uwp`

GitHub	github.com/RoguePlanetoid/Spotify-Uwp
Git	`$ git clone https://github.com/RoguePlanetoid/Spotify-Uwp.git`

Example
using Spotify.Uwp; using Spotify.Uwp.ViewModels; var client = SpotifySdkClientFactory.CreateSpotifySdkClient("client-id","client-secret"); GridView.ItemSource = new ListAlbumViewModel(SpotifySdk.Instance.SpotifySdkClient, AlbumType.NewReleases);

Universal Windows Platform SDK

ISpotifySdkClient

ISpotifySdkClient
GetCategoryAsync
GetArtistAsync
GetAlbumAsync
GetPlaylistAsync
GetTrackAsync
GetAudioAnalysisAsync
GetUserAsync
GetCurrentUserAsync

| ISpotifySdkClient |
| --- |
| ListCategoriesAsync |
| ListArtistsAsync |
| ListAlbumsAsync |
| ListPlaylistsAsync |
| ListRecommendationGenresAsync |
| ListTracksAsync |
| ListAudioFeatureAsync |
| ListAudioFeaturesAsync |

List View Models
ListAlbumViewModel
ListArtistViewModel
ListAudioFeatureViewModel
ListCategoryViewModel
ListFavouriteViewModel
ListPlaylistViewModel
ListRecommendationViewModel
ListTrackViewModel

Universal Windows Platform SDK
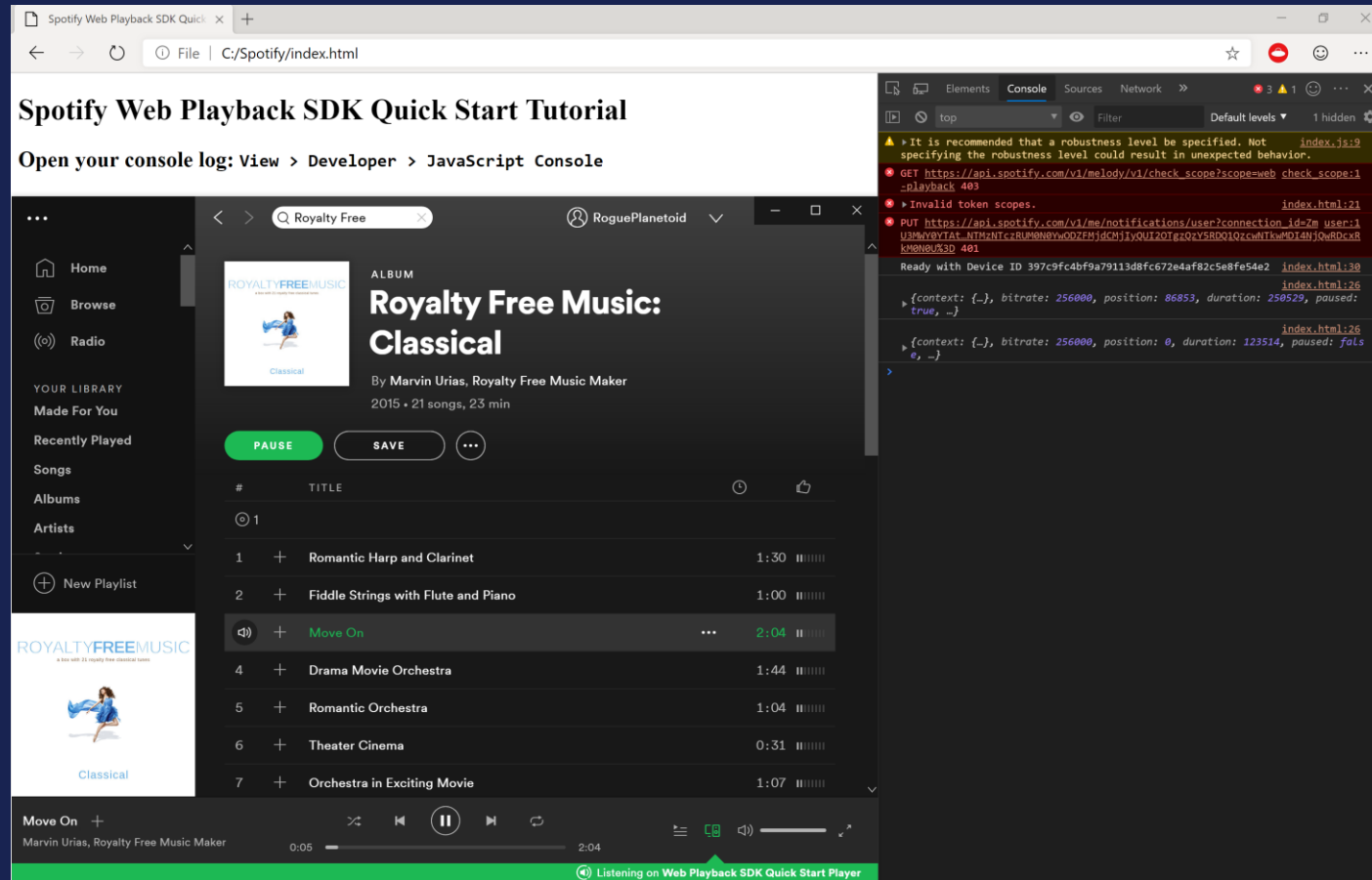
Web Playback SDK – Microsoft Edge   Demo

```csharp
private readonly ISpotifyClient client = SpotifyClientFactory.CreateSpotifyClient(
    "6a53fb50cb7949709041bdc74c75c044", "b9e7de1be14946e0ad080c0c417266aa");
private readonly Uri redirect_url = new Uri("http://example.com/callback");
private const string state = "DDD";

5 references
public enum Option
{
    NewReleases,
    FollowedArtists
}

0 references | 0 exceptions
public MainPage()
{
    this.InitializeComponent();
}

1 reference | 0 exceptions
private void Display_Loaded(object sender, RoutedEventArgs e) =>
    WebView.Navigate(client.AuthUser(redirect_url, state,
    new Requests.Scope { FollowRead = true, LibraryRead = true }));

1 reference | 0 exceptions
private async void WebView_NavigationCompleted(WebView sender,
    WebViewNavigationCompletedEventArgs e)
{
    if (e.Uri.Host == redirect_url.Host)
    {
        WebView.Visibility = Visibility.Collapsed;
        if (await client.AuthUserAsync(e.Uri, redirect_url, state) != null)
            Options.ItemsSource = Enum.GetValues(typeof(Option)).Cast<Option>();
    }
}

1 reference | 0 exceptions
private async void Options_SelectionChanged(object sender, SelectionChangedEventArgs e)
{
    var option = (Option)Options.SelectedItem;
    var cursor = new Requests.Cursor() { Limit = 10 };
    var page = new Requests.Page() { Limit = 10 };
    switch (option)
    {
        case Option.NewReleases:
            Display.ItemsSource = (await client.LookupNewReleasesAsync(page: page)).Albums.Items;
            break;
        case Option.FollowedArtists:
            Display.ItemsSource = (await client.AuthLookupFollowedArtistsAsync(cursor: cursor)).Items;
            break;
    }
}
```

.NET Standard SDK – Blank App (Universal Windows)   Demo
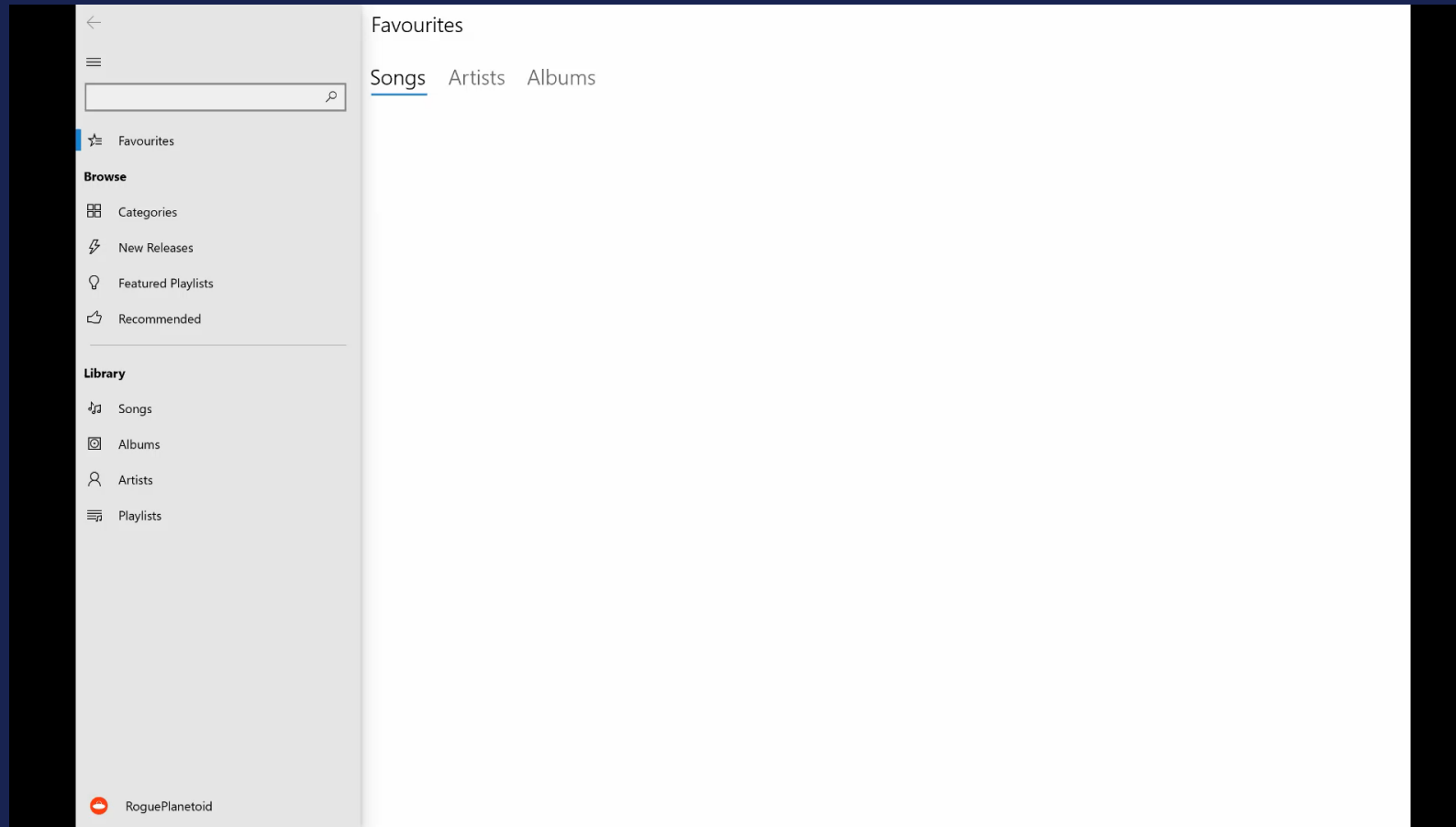
Summary

Spotify®

developer.spotify.com

Summary

Authorisation
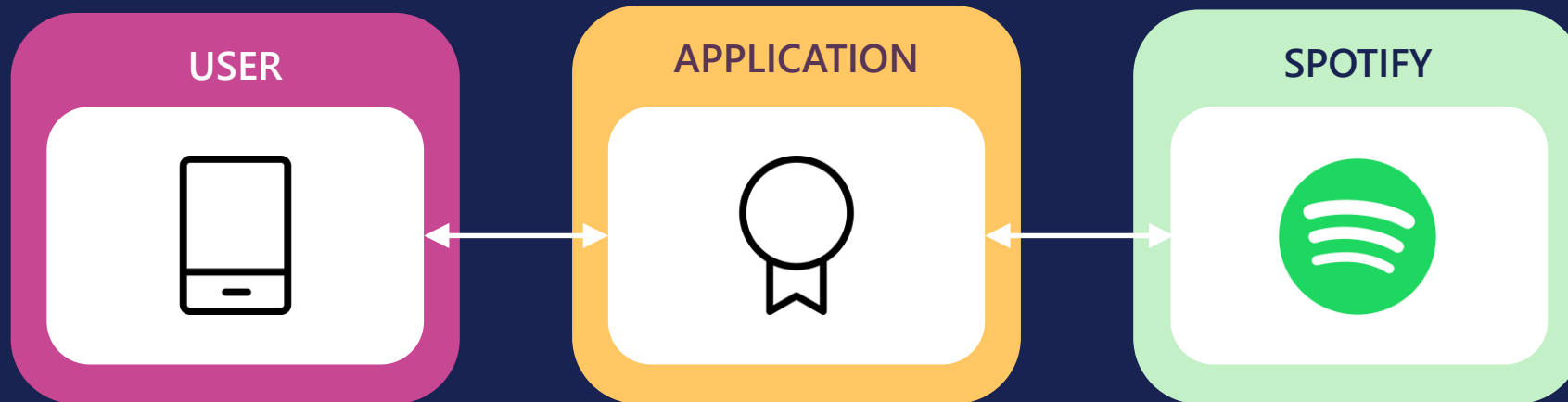
USER

APPLICATION

SPOTIFY

Summary

Spotify Web API

APPLICATION
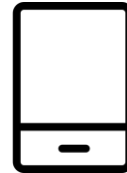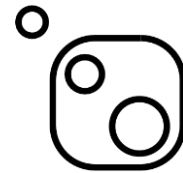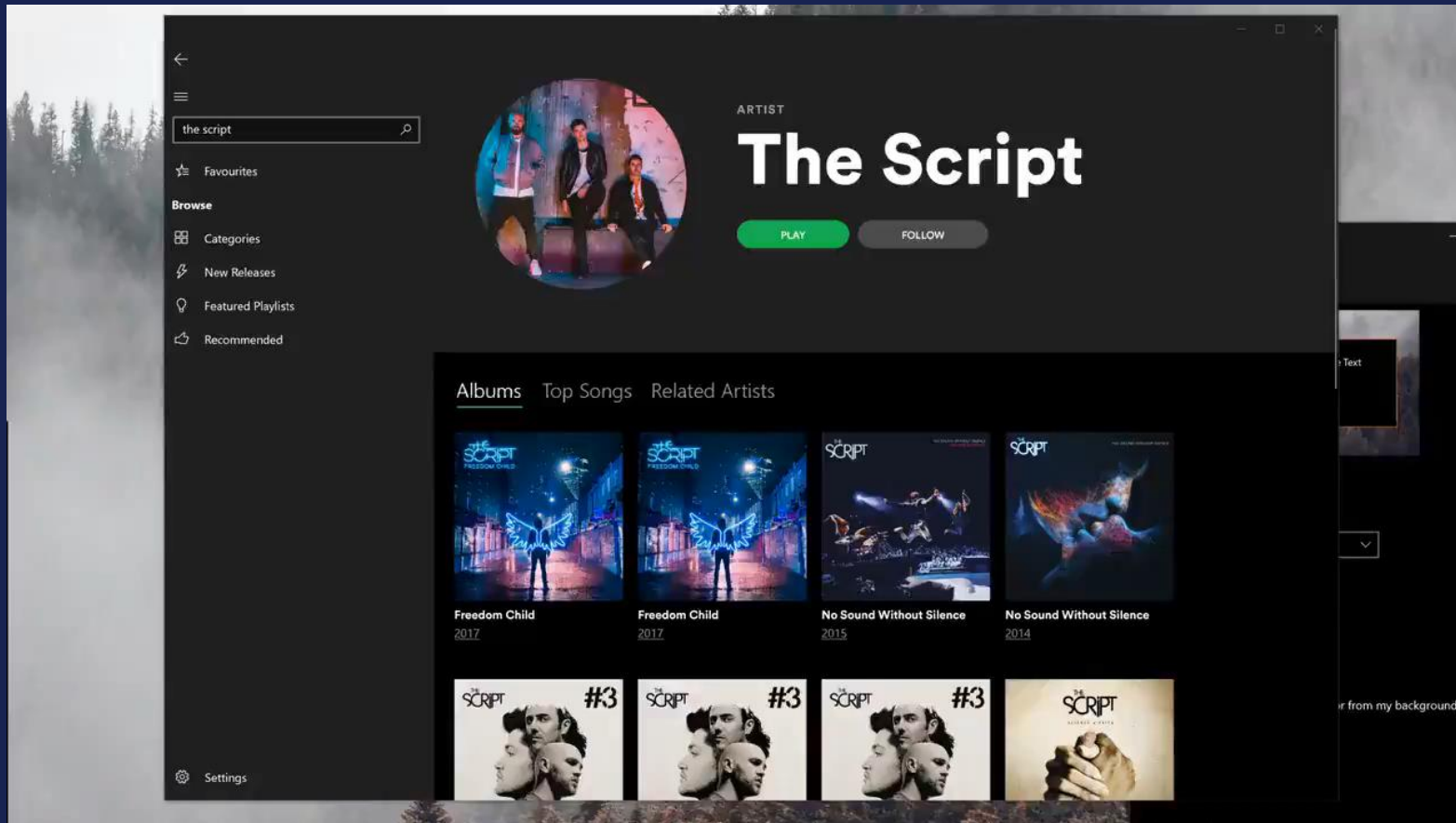
WEB API
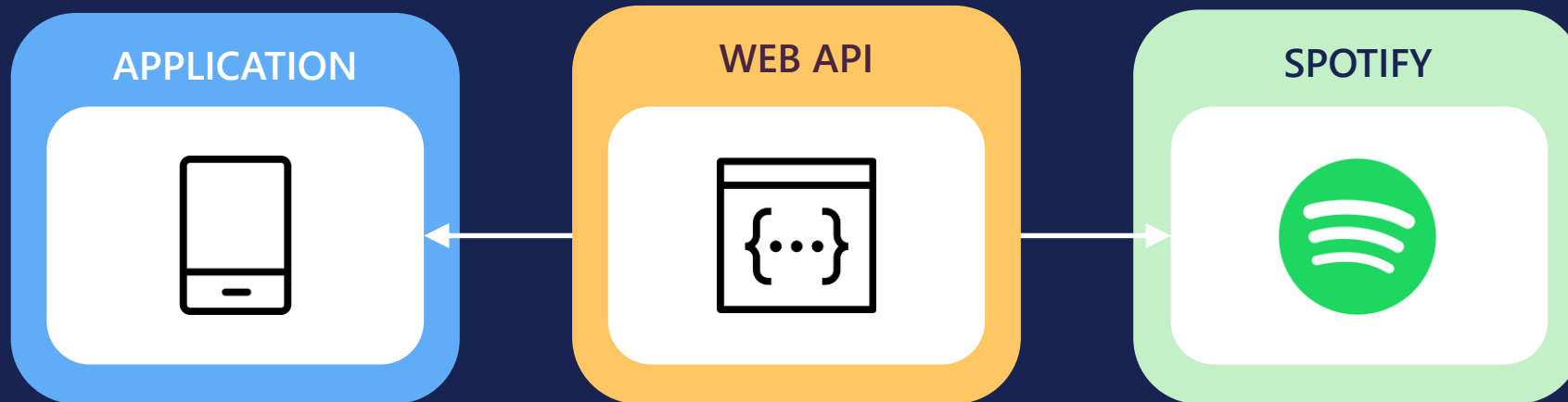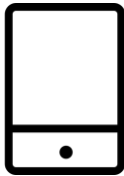
SPOTIFY

Summary

Experiences

Summary

Spotify for Developers

Peter Bull

@rogueplanetoid

comentsys.wordpress.com

github.com/rogueplanetoid

developer.spotify.com

Questions?

Spotify®