

# Music Information and Song Retrieval Web Service

A WCF implementation using GrooveShark API

4/5/2013

Rochester Institute of Technology : CS Department

Karan Moodbidri

## Table of Contents

Introduction .....	3
Technologies .....	3
Newtonsoft Json.NET .....	3
Usage of API .....	4
Single Song Search .....	4
Single Song Metadata Search .....	4
Multiple Song Metadata Search with limit .....	4
Class Structure .....	5
SingleSong.cs .....	5
SingleSongMetadata.cs .....	5
MultipleSongMetadata.cs .....	6
IService1.cs .....	6
Service1.svc.cs .....	6

## Introduction

The Music Information and Song Retrieval Web Service has been realized using the Windows Communication Foundation (WCF) version .NET 4.5.

Path to Create a WCF project:

1. Open Visual Studio
2. Click File
3. Click New
4. Click Project
5. Select WCF Service Application from list of available projects for Visual C#
6. Give Name for Project Below

This is how you create the WCF project.

In the Solution Explorer in the right hand side of Visual Studio you will see the files generated for the project.

## Technologies

The public API i.e. GrooveShark API was called using the URL call. This was achieved by using the 'WebClient' class of System.Net. The values returned from the public API were in JSON i.e. (JavaScript Object Notation ) format. To parse the JSON, a publically available JSON extension for .NET framework i.e. JSON.Net was used. Classes were created to represents segments of the JSON and were parsed and filled in the objects respectively.

## Newtonsoft Json.NET

JSON.Net is used to parse the JSON String. It can be achieved by using the JObject notation or by creating an JArray which contains individual sections of the JSON string.

```
using (WebClient wc = new WebClient())
{
    string json = wc.DownloadString(apiCall);
    var jArray = JArray.Parse(json);
}
```

# Usage of API

## Single Song Search

```
http://tinysong.com/a/UserQuery?format=json&key=APIKey
```

### Input Parameters :

UserQuery : The Query submitted by the user.

APIKey : The key required to call the API

This API call will return the url to the song which is being searched by the user using the userQuery option.

## Single Song Metadata Search

```
http://tinysong.com/b/UserQuery?format=json&key=APIKey
```

### Input Parameters :

UserQuery : The Query submitted by the user.

APIKey : The key required to call the API

This API call will return the metadata of the song which is being searched by the user using the userQuery option.

## Multiple Song Metadata Search with limit

```
http://tinysong.com/b/UserQuery?format=json&limit=userlimit&key=APIKey
```

### Input Parameters :

UserQuery : The Query submitted by the user.

APIKey : The key required to call the API

userLimit : The number of results required by the user in one page.

This API call will return the metadata of all the songs matching with the query searched by the user using the userQuery option.

## Class Structure

### SingleSong.cs

Fields :

1. Url

Use of the class :

The class is used to fill the data returned from the public API into its fields and be serialized and passed across as a customized JSON or XML.

### SingleSongMetadata.cs

Fields :

1. Url
2. SongID
3. SongName
4. artistName
5. albumID
6. albumName

Use of the class :

The class is used to fill the data i.e. metadata about a particular song returned from the public API into its fields and be serialized and passed across as a customized JSON or XML.

### **MultipleSongMetadata.cs**

Fields :

1. List<SingleSongMetadata>()

Use of the class :

The class is used to fill the data i.e. metadata about many songs returned from the public API into the fields of a SingleSongMetadata object and in turn added to this list. This MultipleSongMetadata will be then serialized and passed across as a customized JSON or XML.

### **IService1.cs**

Contains the OperationContracts i.e. the definition of the possible set of operations that can be performed using this web service.

### **Service1.svc.cs**

Contains the implantations of the OperationContracts i.e. the implemantations of the possible set of operations that can be performed using this web service.