

Corso di Programmazione Web e Mobile

A.A. 2018-2019

SPOTIFY DATA

◀ Biagio, Caracciolo, 872522 ▶

Autore:

Biagio Caracciolo

Spotify Data

Accedi con sicurezza ai tuoi dati Spotify

1 Introduzione

Spotify Data è un semplice applicativo server-side che sfrutta la web API di Spotify Developers. Attraverso questa web API l'applicazione fornisce i dati relativi all'utente che ha eseguito l'accesso tramite il suo account (premium o no) di Spotify. L'obiettivo del sito è quello di fornire una base stabile per l'accesso autorizzato ai dati degli account Spotify e alla web API in vista di sviluppi di applicativi che ne necessitano. Le tecnologie usate sono HTML5/CSS, JavaScript, AJAX e Node.js, ed è stata registrata sul sito per sviluppatori Spotify <https://developer.spotify.com/>.

1.1. Breve analisi dei requisiti

1.1.1 Destinatari

Capacità e possibilità tecniche.

L'applicazione può essere utilizzata da chiunque abbia un account Spotify sia gratuito che premium. L'uso è semplicissimo, grazie alla web API di Spotify basterà effettuare l'accesso da web tramite l'interfaccia del sito di Spotify per poi avere a disposizione tutti i dati del proprio account. L'applicazione è studiata per essere usata su qualsiasi device in grado di connettersi ad internet senza richieste di banda particolari.

Linguaggio.

Non è richiesta particolare conoscenza della lingua inglese a parte pochi termini tecnici ormai utilizzati frequentemente da tutti. Ho preferito usare la lingua italiana per l'interfaccia dell'applicazione ma è tranquillamente traducibile in inglese e in altre lingue.

Motivazione.

I destinatari dell'applicativo sono utenti esperti o no che vogliono vedere i propri dati Spotify in un modo più chiaro e unito.

Il sito è anche una base solida che può essere utilizzata per lo sviluppo di altre applicazioni che potrebbero interfacciarsi con Spotify.

1.1.2 Modello di valore

I punti di forza di questa applicazione sono sicuramente la sua semplicità e la sua velocità date dall'interfaccia di accesso di Spotify in modo da mettere l'utente a proprio agio con cose che usa tutti i giorni. L'utente può trovare comodo e semplice il sito con tutti i dati di cui ha bisogno senza perdersi nelle varie pagine di Spotify che potrebbero risultare dispersive specie per chi ha meno dimestichezza con app e web.

È difficile valutare il valore di questo prodotto (denaro, attenzione utenti ecc..) in quanto, a parere mio, è solo la base di quello che potrebbero essere applicazioni più grandi. Ad esempio un sito riguardante musica potrebbe utilizzarla per far registrare gli utenti direttamente con Spotify in modo da avere a disposizione tutti i dati personali, i gusti musicali, playlist ecc..

1.1.3 Flusso dei dati

I contenuti vengono proposti nella maniera più semplice possibile. I dati vengono scambiati tra client e server attraverso un oggetto JSON contenenti tutte le informazioni necessarie e poi vengono rielaborate attraverso un JavaScript. La chiamata è effettuata anche con il token dell'app (fornito dopo averla registrata su Spotify Developers) che permette di autenticare gli account tramite la pagina di login di Spotify.

Ottenere i contenuti.

I contenuti sono disponibili graduata mente grazie alla web API di Spotify Developers, basta registrare l'applicazione sull'apposita dashboard.

Archiviare e organizzare i contenuti

Non si archivia alcun tipo di contenuto, gli oggetti JSON forniscono dati che vengono mandati in output dalla parte JavaScript.

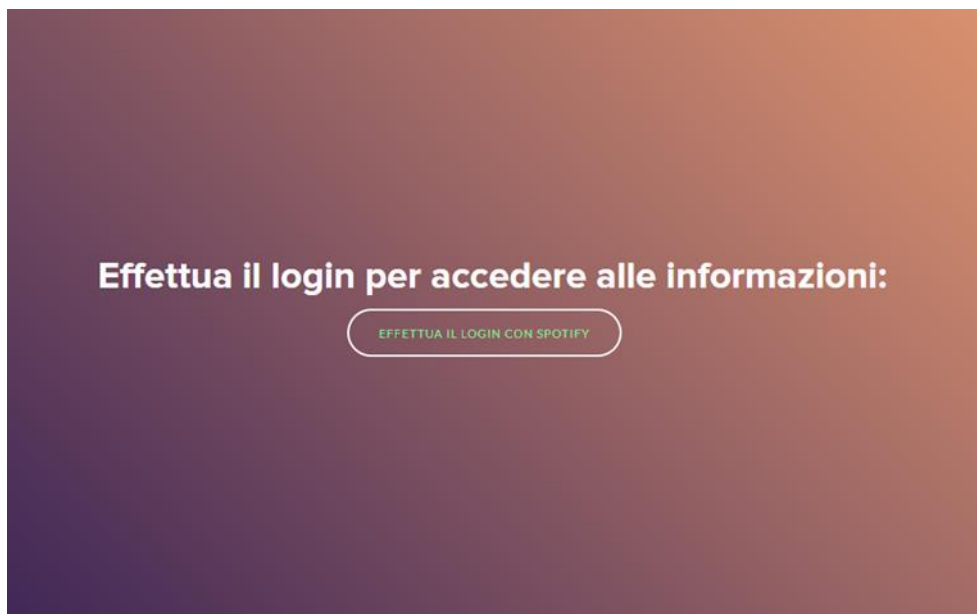
Pubblicare i contenuti

I contenuti vengono reperiti come stringhe quindi ho creato un output personalizzato in modo da mostrare i dati all'utente in modo intuitivo.


I contenuti sono forniti dall'api quindi non c'è bisogno di aggiornamenti da parte dello sviluppatore, mentre la web API Spotify è completamente gestita esternamente.

Non ci sono vincoli alla pubblicazione, l'immagine di sfondo è stata fatta da me e il CSS che ho scelto di usare è gratuito. Anche la web API è completamente gratuita.

2 Interfacce



Questa è la schermata principale, è molto semplice, il bottone porta direttamente alla schermata di login di Spotify. Ho cercato di tenere un layout simile a quello di Spotify lavorando su carattere e colori.



Per continuare, accedi a Spotify.

ACCEDI CON FACEBOOK

o

Indirizzo e-mail o username

Inserisci il tuo nome utente Spotify o il tuo indirizzo e-mail.

Password

Inserisci la tua password.

☒ Ricordami

ACCEDI

Hai dimenticato la password?

Non hai un account?

ISCRIVITI A SPOTIFY

Se clicchi su "Accedi con Facebook" e non sei un utente Spotify, sarai registrato su Spotify e accetterai i relativi [Termini e condizioni](#) e [Informativa sulla privacy](#).

Questa è la schermata di login di Spotify a cui si viene reindirizzati per effettuare il login dal sito. Questa schermata non verrà visualizzata se nella sessione del browser si è già loggati su Spotify.

Benvenuto/a Biagio Caracciolo questi sono i tuoi dati Spotify:



Display name	Biagio Caracciolo
Data di nascita	1996-09-05
Id	q325gmipviiiolyrtdyhkon9
Email	bigius92@gmail.com
Spotify URI	https://open.spotify.com/user/q325gmipviiiolyrtdyhkon9
Immagine profilo	https://profile-images.scdn.co/images/userprofile/default/a6e7c6ae661a1bfa263335ad1dd5707aae6d7837
Paese	IT

oAuth info

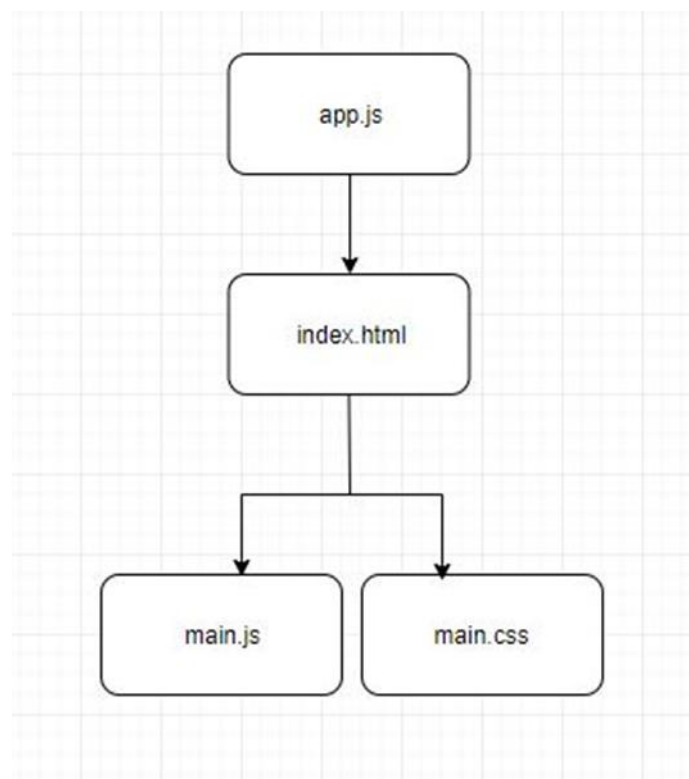
Access token	BQB5SmOE_Z5EJZDC9cuQdMhRbmVCFMuWal2Mb1MNnPbuKMjjBFteluMKt-ffJFFK2RdGSEN85Euq3rqswp...
Refresh token	AQBXEJHcdZ7osTX8y1SwivvPg3lpyNk-T67IT-JR4N0KRFy_R74W30zu9_XE9WqJIEkWiFwTqdU4p98S2u7xzT...

OTTIENI UN NUOVO TOKEN

Questa invece è la schermata finale. Questo è un esempio di come i dati del mio account Spotify vengono mostrati: Display name, data, email ecc... Sono anche presenti due link che riportano rispettivamente all'URI di Spotify (Il lettore musicale web) e ad un link univoco dell'immagine del profilo. È mostrato anche l'access token dell'account che è univoco per l'account e tramite il bottone c'è la possibilità di refresharlo grazie ad uno script.

3 Architettura

3.1 Diagramma dell'ordine gerarchico delle risorse



4 Codice

4.1 HTML

4.1.1 body

```
<body>
  <div class="container">
    <div id="login">
      <h1 style="text-align:center;">Effettua il login per accedere alle informazioni:</h1>
      <div class="wrap"><button class="btn btn-conf btn-clear"><a href="/login">Effettua il login con Spotify</a></button></div>
    </div>
    <div id="loggedin">
      <div id="user-profile">
        </div>
      <div id="oauth">
        </div>
      <div class="wrap"><button class="btn btn-conf btn-clear" id="obtain-new-token">Ottieni un nuovo token</button></div>
    </div>
  </div>
```

4.1.2 output

```
<script id="user-profile-template" type="text/x-handlebars-template">
  <h1>Benvenuto/a {{display_name}} questi sono i tuoi dati Spotify:</h1>
  <div class="media">
    <br>
    <br>
    <br>
    <div class="pull-left">
      
    </div>
    <div class="media-body">
      <dl class="dl-horizontal">
        <dt>Display name</dt><dd class="clearfix">{{display_name}}</dd>
        <dt>Data di nascita</dt><dd class="clearfix">{{birthdate}}</dd>
        <dt>Id</dt><dd>{{id}}</dd>
        <dt>Email</dt><dd>{{email}}</dd>
        <dt>Spotify URI</dt><dd><a href="{{external_urls.spotify}}">{{external_urls.spotify}}</a></dd>
        <dt>Immagine profilo</dt><dd class="clearfix"><a href="{{images.0.url}}">{{images.0.url}}</a></dd>
        <dt>Paese</dt><dd>{{country}}</dd>
      </dl>
    </div>
  </div>
</script>

<script id="oauth-template" type="text/x-handlebars-template">
  <br>
  <br>
  <br>
  <h2>oAuth info</h2>
  <dl class="dl-horizontal">
    <dt>Access token</dt><dd class="wrap text-overflow">{{access_token}}</dd>
    <dt>Refresh token</dt><dd class="wrap text-overflow">{{refresh_token}}</dd>
  </dl>
</script>
```

4.2 CSS

```
* {
  margin: 0;
  padding: 0px;
  font-family: 'Lato', sans-serif;
}

body {
  background-image: url("bg.jpg");
  margin: 0;
  color: #ffffff;
}

h1, h2, h3, h4, h5, h6 {
  font-family: 'font';
  font-weight: 300;
  text-align: left;
}

p {
  padding: 0;
  margin-bottom: 12px;
  font-family: 'Lato', sans-serif;
  font-weight: 300;
  font-size: 18px;
  line-height: 28px;
  color: #ffffff;
  margin-top: 10px;
}

body {
  height: 100%;
}
```

```
.btn-conf {
  border-radius: 50px;
  margin-right: 15px;
  font-size: 11px;
  letter-spacing: 1px;
  text-transform: uppercase;
  padding: 14px 28px 13px 28px;
}

.btn-clear {
  background: transparent;
  color: white;
  border: 2px solid white;
}

.btn-clear:hover {
  background: white;
  color: #2f2f2f;
}
```

```
@font-face {
  font-family: font;
  src: url(font.otf);
}
```


4.3 API

4.3.1 acquisizione dei dati

```
function getHashParams() {
  var hashParams = {};
  var e, r = /(?:^&=)=[^&]*&?/g,
      q = window.location.hash.substring(1);
  while ( e = r.exec(q)) {
    hashParams[e[1]] = decodeURIComponent(e[2]);
  }
  return hashParams;
}

var userProfileSource = document.getElementById('user-profile-template').innerHTML,
    userProfileTemplate = Handlebars.compile(userProfileSource),
    userProfilePlaceholder = document.getElementById('user-profile');

var oauthSource = document.getElementById('oauth-template').innerHTML,
    oauthTemplate = Handlebars.compile(oauthSource),
    oauthPlaceholder = document.getElementById('oauth');

var params = getHashParams();

var access_token = params.access_token,
    refresh_token = params.refresh_token,
    error = params.error;

if (error) {
  alert("Errore durante l'autenticazione");
} else {
  if (access_token) {
    // render oauth info
    oauthPlaceholder.innerHTML = oauthTemplate({
      access_token: access_token,
      refresh_token: refresh_token
    });
  }
}
```


4.1.1 autorizzazioni

```
$.ajax({
  url: 'https://api.spotify.com/v1/me',
  headers: {
    'Authorization': 'Bearer ' + access_token
  },
  success: function(response) {
    userProfilePlaceholder.innerHTML = userProfileTemplate(response);

    $('#login').hide();
    $('#loggedin').show();
  }
});
} else {
  // render initial screen
  $('#login').show();
  $('#loggedin').hide();
}

document.getElementById('obtain-new-token').addEventListener('click', function() {
  $.ajax({
    url: '/refresh_token',
    data: {
      'refresh_token': refresh_token
    }
  }).done(function(data) {
    access_token = data.access_token;
    oauthPlaceholder.innerHTML = oauthTemplate({
      access_token: access_token,
      refresh_token: refresh_token
    });
  });
}, false);
```

4.4 Node.js

```

var express = require('express'); // Express web server framework
var request = require('request'); // "Request" library
var cors = require('cors');
var querystring = require('querystring');
var cookieParser = require('cookie-parser');

var client_id = 'ed0ef64ee950473d8bfc611de0ee6c09'; //client id
var client_secret = '1806dd9da0554a1d820b9261218fc973'; //secret
var redirect_uri = 'http://localhost:8888/callback/'; // redirect uri

/**
 * Generates a random string containing numbers and letters
 * @param {number} length The length of the string
 * @return {string} The generated string
 */
var generateRandomString = function(length) { ...
};

var stateKey = 'spotify_auth_state';

var app = express();

app.use(express.static(__dirname + '/public'))
  .use(cors())
  .use(cookieParser());

app.get('/login', function(req, res) { ...
});

app.get('/callback', function(req, res) { ...
});

app.get('/refresh_token', function(req, res) { ...
});

console.log('Listening on 8888');
app.listen(8888);

```

5 Conclusioni

L'applicazione é semplice e veloce, potrebbe essere commercializzabile tranquillamente da sola ma credo che potrebbe dare il massimo se integrata a servizi già più grandi e famosi che vorrebbero utilizzare i dati e le interfacce di Spotify, l'app di musica più famosa e utilizzata al momento. A mio parere l'applicazione grazie alla sua semplicità potrebbe contornare anche servizi mobile.

6 Nota bibliografica e sitografica

- <https://developer.spotify.com/>
- <https://developer.spotify.com/documentation/>
- <https://www.w3schools.com/>
- <https://fontsfree.net/freefonts/>
- <https://nodejs.org/it/docs/>