



Projet - Deez Web

Développement Web - Brief

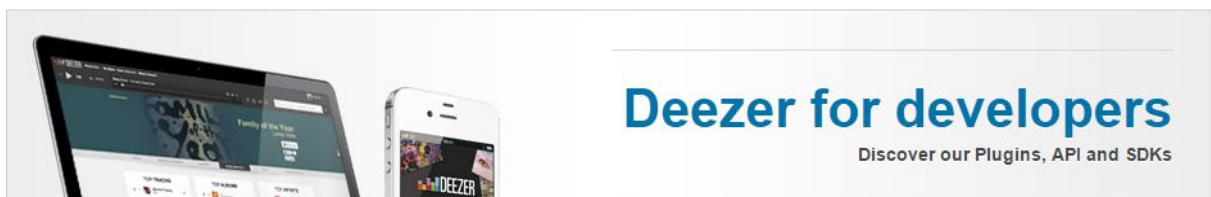
Objectif

Vous devez réaliser une application web permettant de rechercher, explorer et organiser des **musiques** en vous aidant de l'API développeurs gratuite fournie par le service Deezer.

Elle sera constituée d'une fonctionnalité de recherche de musiques, et d'une fonctionnalité de mise en favoris pour les musiques préférées.

Les informations des favoris seront stockées directement dans le navigateur (via l'API « localStorage »).

Le service (API) Deezer



L'API que vous utiliserez sera celle de Deezer : <https://developers.deezer.com>



Un compte Deezer gratuit est nécessaire pour accéder aux données.

Une fois votre compte créé, logguez-vous via le bouton **Login**.

Une fois loggué, vous accéderez au panneau suivant :

Introduction

Global parameters

API errors

AUTHENTICATION

OAuth

Permissions

TOOL

API Explorer

oEmbed

API OBJECTS

Album

Artist

Chart

Comment

Editorial

Episode

Genre

Infos

Options

Playlist

Podcast

Radio

Search

Track

User

ACTIONS

Create/Edit (POST)

Delete (DELETE)

Introduction

Unlimited Access, without stress, without identification. Deezer Simple API provides a nice set of services to build up web applications allowing the discovery of Deezer's music catalogue.

Requests

You may use ordinary HTTP GET messages. The base URL for each API method looks like the following

```
https://api.deezer.com/version/service/id/method/?parameters
```

Query quota

The number of requests per second is limited to 50 requests / 5 seconds.

Requests format examples

```
https://api.deezer.com/user/2529
```

```
https://api.deezer.com/user/2529/playlists
```

```
https://api.deezer.com/album/302127
```

Response format

In order to search the specified artist name 'eminem', and get xml back, you can do :

```
GET /search/artist?q=eminem&index=0&limit=2&output=xml
```

The available formats are those following :

DESCRIPTION	ACCEPT HEADER	EXTENSION
JSON	application/json	.json
JSONP	application/json	.json
XML	application/xml, text/xml	.xml

L'API de Deezer est très riche et permet de nombreuses choses. Néanmoins, l'élément principal qui va vous intéresser est celui encadré en vert sur la capture, à savoir le endpoint de recherche « **Search** ».

Pages

Le projet que vous devez réaliser est une **application web utilisant les informations de l'API Deezer**. Votre application doit être constituée de 3 pages au total :

- Une page d'accueil
- Une page de recherche
- Une page listant les musiques favorites de l'utilisateur

Détails et instructions pour chaque page

Les visuels qui suivent sont présents à des fins d'illustration. **Il ne s'agit pas de maquettes à intégrer.** Vous restez maître de l'aspect visuel, de l'agencement des informations et de l'UX (expérience utilisateur) de votre interface finale !

Vous devez faire preuve de bon sens et de créativité pour présenter au mieux les informations sur chaque page.

NB : Ces visuels sont également disponible au format PNG, et joints avec ce brief

1. Page d'accueil

Une simple page de bienvenue sur l'application, qui proposera également au visiteur une musique à écouter au hasard **parmis ses favoris**. Si le visiteur n'a encore aucun favoris, alors le bloc ne s'affichera pas.:

Captures d'écran :

Si l'utilisateur n'a encore aucun favori :

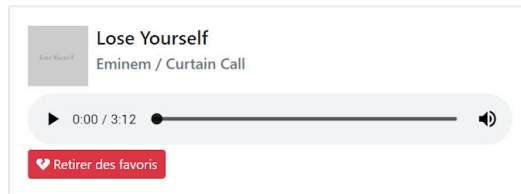


Si l'utilisateur a déjà un ou plusieurs favoris :

Bienvenue sur Deezweb

L'application qui permet facilement d'écouter de la musique.

Une musique de vos favoris au hasard:



[Choisir une autre musique](#)

2. Page de recherche

La page doit être composée d'un formulaire avec au minimum 2 champs :

- Un champs pour rechercher un titre ("q")
- Un champs pour préciser le mode de tri ("order") et permettant de trier sur 5 critères :
 - par Album (croissant)
 - par Artiste (croissant)
 - par Titre (croissant)
 - par popularité (croissant)
 - par rang (croissant)

Ces deux paramètres peuvent être passés à l'API de Deezer de la façon suivante :

https://api.deezer.com/search?q=TITRE+A=RECHERCHER&order=RATING_ASC

Le paramètre « ?q= » correspond à la recherche et le paramètre « &order= » permet de trier les résultats.

Plus de détails sur le endpoint "Search" de l'API Deezer :

<https://developers.deezer.com/api/search>

Les résultats de la recherche doivent être présentés sous forme de liste de musiques (tracks) et afficher pour chaque résultat les informations suivantes :

- Titre de la musique
- Nom de l'artiste
- Nom de l'album
- Un player audio permettant de jouer la musique.
- Un bouton permettant soit d'ajouter la musique dans les favoris, ou de l'en supprimer si elle s'y trouve déjà.

Si la recherche de l'utilisateur ne donne aucun résultat sur l'API de Deezer, un message devra s'afficher à la place.

Captures d'écran :

Formulaire vierge :

Deezweb

Accueil

Rechercher un titre

Mes favoris

Recherche

Recherchez un titre sur Deezer en utilisant le formulaire suivant :

Titre :

Eminem, Armin Van Buuren, Rihanna, ...

Trier par :

Album

Album

Artiste

Musique

Les plus populaires

Les mieux notés

Go

Avec résultats :

Recherche

Recherchez un titre sur Deezer en utilisant le formulaire suivant :

Titre :

Trier par :

Album

Go

Résultats



Lose Yourself
Eminem / Curtain Call

▶ 0:00 / 3:12

🔊

Retirer des favoris



The Real Slim Shady
Eminem / Curtain Call

▶ 0:00 / 3:12

🔊

Ajouter aux favoris



Without Me
Eminem / The Eminem Show

▶ 0:00 / 3:12

🔊

Ajouter aux favoris

Sans résultats :

Recherche

Recherchez un titre sur Deezer en utilisant le formulaire suivant :

Titre :

Trier par :

Artiste



Go

Aucun résultat pour cette recherche ...

3. Page de favoris

Une page listant les favoris que le visiteur a ajouté durant son utilisation de l'application.

Il doit être possible depuis cette page de retirer un favori.

Les données doivent être stockées dans le **localStorage** du navigateur web.

Un rappel vers le lien d'Alsacrérations expliquant en détails le fonctionnement de l'API Web Storage :

<https://www.alsacreations.com/article/lire/1402-web-storage-localstorage-sessionstorage.html>

Captures d'écran :

Sans favoris :

Mes favoris



Liste de mes titres favoris

Aucun favoris dans votre liste ...

Avec favoris :

Mes favoris

Liste de mes titres favoris

 <p>Lose Yourself Eminem / Curtain Call</p> <p>▶ 0:00 / 3:12</p> <p>♥ Retirer des favoris</p>	 <p>Venom Eminem / Venom Single</p> <p>▶ 0:00 / 3:12</p> <p>♥ Retirer aux favoris</p>
---	---

Modalités du projet

Ce projet est à réaliser sur une période d'une semaine, en centre de formation, en présence d'un intervenant assurant le suivi technique du projet.

- Vous devez **obligatoirement être présent** durant les heures de suivi.
- La communication et l'échange entre les étudiants est encouragé, cependant le travail à fournir reste **individuel**.

Technologies

Vous devez utiliser vos connaissances en HTML, CSS et JavaScript pour réaliser ce projet.

L'utilisation de la bibliothèque JavaScript jQuery n'est pas obligatoire, mais recommandée.

Versionning

Vous utiliserez Git pour versionner votre projet, et ferez des commits cohérents et réguliers pour sauvegarder votre travail.

Rendus et mise en ligne

Le rendu s'effectue le Vendredi en fin de journée, et se fait sur Github.

Critères de notation

Seront pris en compte les critères suivant pour la note du projet :

- Application fonctionnelle (respect des fonctionnalités demandés sur chacune des 3 pages)
(9 points)
- Maîtrise de la bibliothèque (ou du framework) JavaScript utilisé
(4 points)
- Utilisation conforme d'AJAX et de l'API Deezer
(3 points)
- Ergonomie et qualité de l'interface
(2 points)
- Qualité du code HTML, CSS et JavaScript (respect des normes W3C)
(2 points)

Total sur **20 points**.

Points bonus :

Des points bonus pourront être attribués à la discrétion du formateur sur les critères suivants:

- Prise d'initiative sur l'amélioration des fonctionnalités demandées.
- Ajout d'animations améliorant l'ergonomie et l'interface.
- Gestion des erreurs :

- Dans le cas où l'application n'a plus de réseau
- Dans le cas où un identifiant d'artiste n'existe pas
- ...

De la même manière, des points pourront être retirés en cas de constat de triche évidente, de non-présence durant les heures de cours, ou de mauvais comportement.