

Miled
Yousri
1R3

Rapport PFMP N°1

Du 21 novembre au 16 décembre



Baccalauréat Professionnel Système Numérique
Option: Réseau Informatique Système Communicant

Lycée Gustave Ferrié
7 Rue des Écluses Saint-Martin, 75010 Paris
0142021955

Remerciements	2
Introduction	3
Présentation de l'entreprise	4;5
Compte rendu d'activités	6
Étude de cas	7;8;9
Conclusion	10
<hr/>	
Semaine 1 : Data et Product, langage SQL	11
Présentation de l'équipe recommandation	11
Observation	11
Tâches	12
Requêtes SQL	12
Analyses	12
Semaine 2 : Metadata curation	13
Présentation de l'équipe	13
Observation	13
Tâches	13
Sélections des meilleur titres	13
Ajout des titres dans la playlist, Ajout des covers	13;14
Semaine 3 : Data et Research, langage Python	15
Présentation de l'équipe	15
Observation	15
Tâches	15
Apprentissage du Python	15
Création du Blind test	15;16
Semaine 4 : Royalties et Reporting, manipulation d'API	17
Présentation de l'équipe	17
Observation	17
Tâches	17
Création de l'App	17
Utilisation de l'API	18

Remerciements

Je commence ce rapport de stage en remerciant Adrianna Avizino et Arielle Marouani pour m'avoir trouvé une place de stagiaire chez Deezer et m'avoir aidé pour mon rapport de stage et mon CV. Je remercie M. Giraudeau Lauren pour avoir été attentif et pour m'avoir suivi pendant ce stage.

Je remercie encore Arielle Marouani pour m'avoir appris le langage SQL et pour m'avoir accompagné pendant ce stage. Elle a su m'épauler et me suivre. Je remercie également Deniz Mekik et Fouad Youssef pour m'avoir appris à traiter les metadatas, Darick Lean pour m'avoir appris le langage Python. Je remercie également Leandro Desuque pour m'avoir expliqué le fonctionnement des Royalties.

Introduction

Je suis élève en classe de première au lycée professionnel Gustave Ferrie dans le 10ème arrondissement de Paris, dans le domaine du Réseau Informatique. Dans le cadre de ma formation je dois effectuer deux stages en première et deux stages en terminale.

Mon objectif pendant ce stage est de sortir de cette entreprise avec de nouvelles connaissances informatiques et de compétences et de comprendre le monde de l'entreprise et son fonctionnement.

Présentation de l'entreprise

Deezer est une application de streaming musical qui donne accès à une grande variété de titres et à d'autres contenus audio tels que des podcasts. Le service est intégré aux appareils de plus de 80 marques tech grand public.

Deezer est situé au 24 Rue de Calais, 75009 Paris en face du Square Hector Berlioz. Son numéro de siren est 51171657300072 et son numéro SA est 511 716 573 est son numéro de téléphone est: 01 84 25 25 32.

Situations géographique:

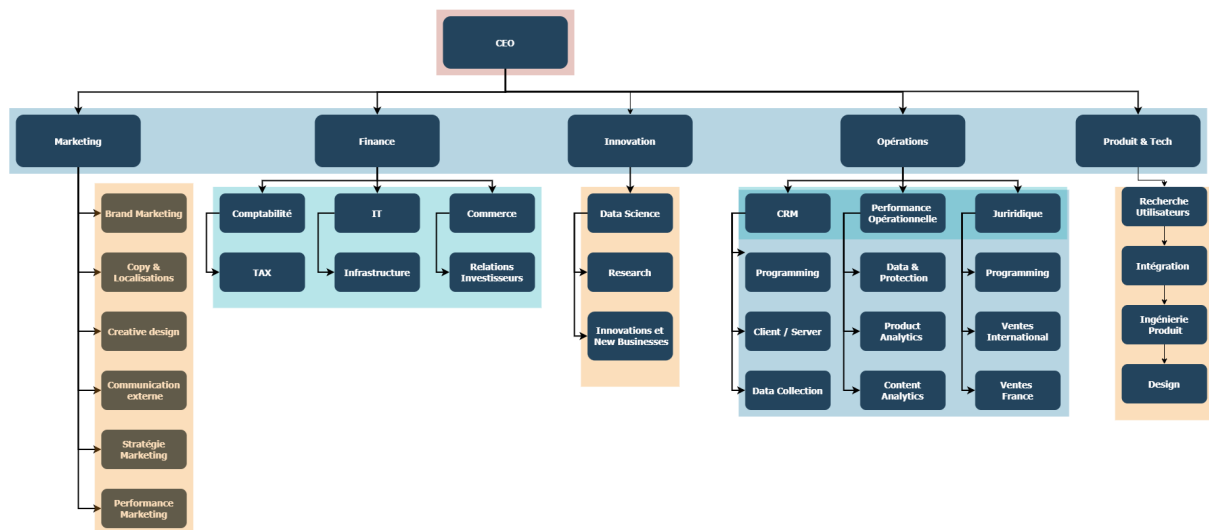


Deezer est basée sur le traitement de données, l'hébergement et les activités connexes.
 Sur l'année 2021 Deezer réalise plus de 400 millions d'euros de chiffre d'affaires soit 29%.
 Sur l'année 2017 Deezer réalise un chiffre d'affaires de 280 766 200.00 €.
 Cette entreprise est basé sur la distribution

Deezer est une plateforme française de distribution numérique dédiée à l'écoute de musique en streaming lancée en août 2007. En octobre 2014, Deezer annonce l'acquisition de **Stitcher**, un fournisseur américain de podcasts et contenus radios Ce service permet l'écoute de musiques et de podcasts, issus d'un catalogue négocié avec les labels regroupant plus de 90 millions de titres en 2022.

Deezer touche tout type de clients, cependant l'entreprise vise un public entre 18 et 24 ans. Les projections futures visent une entrée d'âge à partir de l'adolescence.
 Deezer est en partenariat avec plusieurs marques comme Orange, la Fnac, Cdiscount, Sonos, BOSE, SAMSUNG, ProSiebenSat.Media SE, Cricket et plus encore.

Deezer est en concurrence avec plusieurs marques comme Spotify, Youtube music, Google Play Music et bien encore.



Compte rendu d'activités

Semaine 1 : 21 au 25 Novembre

Lors de la première semaine nous avons eu des TPs à faire sur des requêtes SQL avec Arielle Marouani.

Semaine 2 : 28 Novembre au 2 Décembre

Lors de la deuxième semaine nous avons créé des playlists 100% avec Deniz Mekik, Fouad Youssef et Béranger Magni.

Semaine 3 : 5 au 9 Décembre

Pendant cette troisième semaine nous avons appris les notions du Python pour créer un blind test avec Darick Lean.

Semaine 4 : 12 au 16 Décembre

Pour finir, lors de cette quatrième semaine nous avons travaillé sur la gestion des royalties avec Leandro Desuque.

Étude de cas

Manipulation des APIs

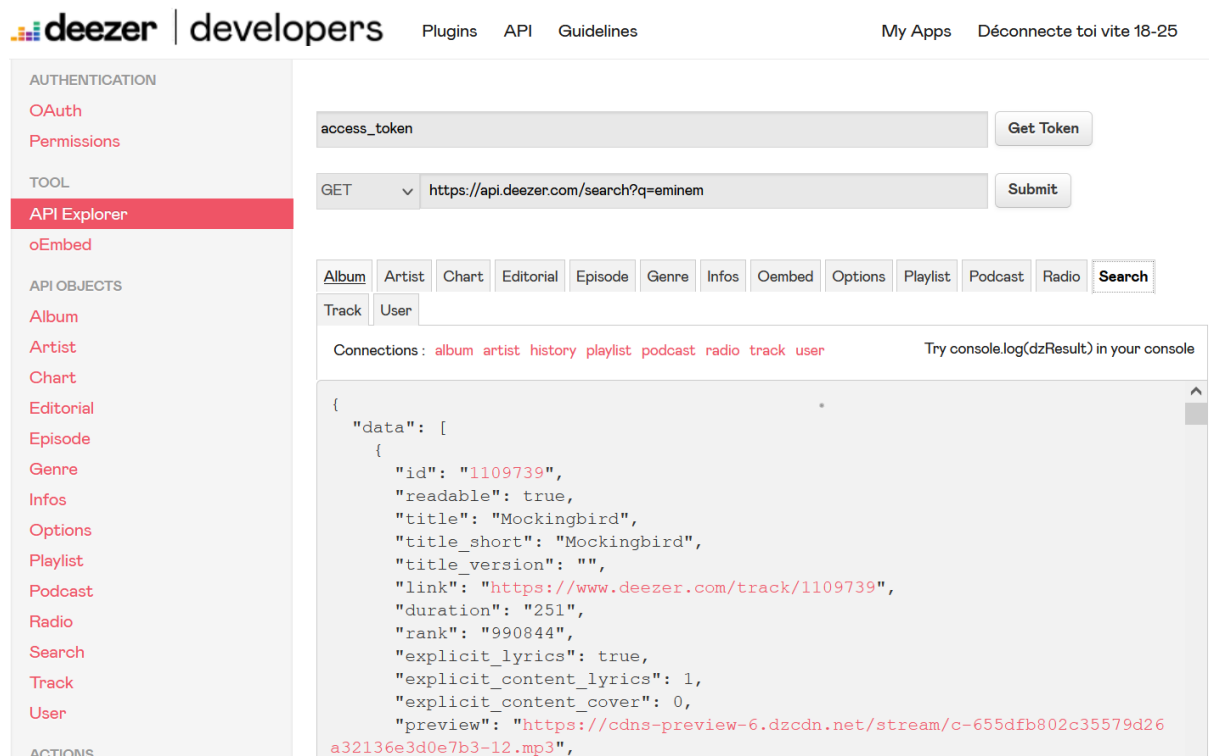
J'ai choisi de concerner mon étude de cas sur la manipulation d'API car je trouve que c'est l'activité la plus intéressante de mon stage.

L'objectif de cette activité était de récupérer plusieurs informations qui nous avaient été demandées auparavant.

Pour réaliser cette activité nous avons eu des documents détaillés sur la gestion des APIs et nous avons aussi eu des documents avec les résultats attendus.

Pour commencer nous nous sommes connectés au site pour développeurs de Deezer, nous avons créé une App avec plein de paramètres qui nous a été envoyé par mail et nous l'avons mis en local, récupérer le token du compte connecté à l'app ensuite nous étions en mesure d'utiliser l'API et de personnaliser les commandes URL selon la tâche qu'il fallait faire.

Le premier jour de cette activité J'ai eu de la difficulté à adapter les commandes d'URL pour obtenir le résultat attendu mais heureusement J'ai pu m'orienter en cherchant les bonnes commandes sur l'application Deezer for developers.



The screenshot shows the Deezer Developers API Explorer interface. On the left, there's a sidebar with navigation links: AUTHENTICATION (OAuth, Permissions), TOOL (API Explorer, oEmbed), API OBJECTS (Album, Artist, Chart, Editorial, Episode, Genre, Infos, Options, Playlist, Podcast, Radio, Search, Track, User), and ACTIONS. The main area shows a GET request to `https://api.deezer.com/search?q=eminem` with a 'Get Token' button. Below the request, there's a 'Connections' section with tabs for album, artist, history, playlist, podcast, radio, track, and user. The 'track' tab is selected, showing a JSON response for the track 'Mockingbird' by Eminem. The response includes details like id, readable status, title, link, duration, rank, explicit lyrics, and a preview URL.

```
{
  "data": [
    {
      "id": "1109739",
      "readable": true,
      "title": "Mockingbird",
      "title_short": "Mockingbird",
      "title_version": "",
      "link": "https://www.deezer.com/track/1109739",
      "duration": "251",
      "rank": "990844",
      "explicit_lyrics": true,
      "explicit_content_lyrics": 1,
      "explicit_content_cover": 0,
      "preview": "https://cdns-preview-6.dzcdn.net/stream/c-655dfb802c35579d26a32136e3d0e7b3-12.mp3",
    }
  ]
}
```

Pour obtenir mes chansons favorites il fallait copier l'URL **track** dans l'application Deezer for developers Coller l'URL dans l'application Postman, ouvrir cmd et demandé l'accès au token Coller l'URL du token dans l'application postman et on obtient le résultat.

Etape 1:

deezer | developers Plugins API Guidelines My Apps Déconnecte toi vite 18-25

OAuth Permissions

TOOL

API Explorer

oEmbed

API OBJECTS

Album

Artist

Chart

Editorial

Episode

Genre

Infos

Options

Playlist

Podcast

Radio

Search

Track

User

access_token

GET

Submit

Try console.log(dzResult) in your console

```
{
  "id": "3135556",
  "readable": true,
  "title": "Harder, Better, Faster, Stronger",
  "title_short": "Harder, Better, Faster, Stronger",
  "title_version": "",
  "isrc": "GBDUW0000059",
  "link": "https://www.deezer.com/track/3135556",
  "share": "https://www.deezer.com/track/3135556?utm_source=deezer&utm_content=track-3135556&utm_term=0_1672154320&utm_medium=web",
  "duration": "224",
  "track_position": 4,
  "disk_number": 1,
  "rank": "807429",
  "release_date": "2005-01-24",
  "explicit_lyrics": false,
  "explicit_content_lyrics": 0,
}
```

Etape 2:

P-Tech

GET name

GET info sur moi

Slack Web API Essential Collectio

Admin API

Web API

GET

Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> access_token	fr5ozosouFGQPypg36rsK2yfuzd8oxzm8A...			
Key	Value	Description		

Body Cookies (1) Headers (21) Test Results

200 OK 352 ms 4.39 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "data": [
3     {
4       "id": 1942219747,
5       "readable": true,
6       "title": "Do It Again (feat. 2Rare)",
7       "link": "https://www.deezer.com/track/1942219747",
8       "duration": 213,

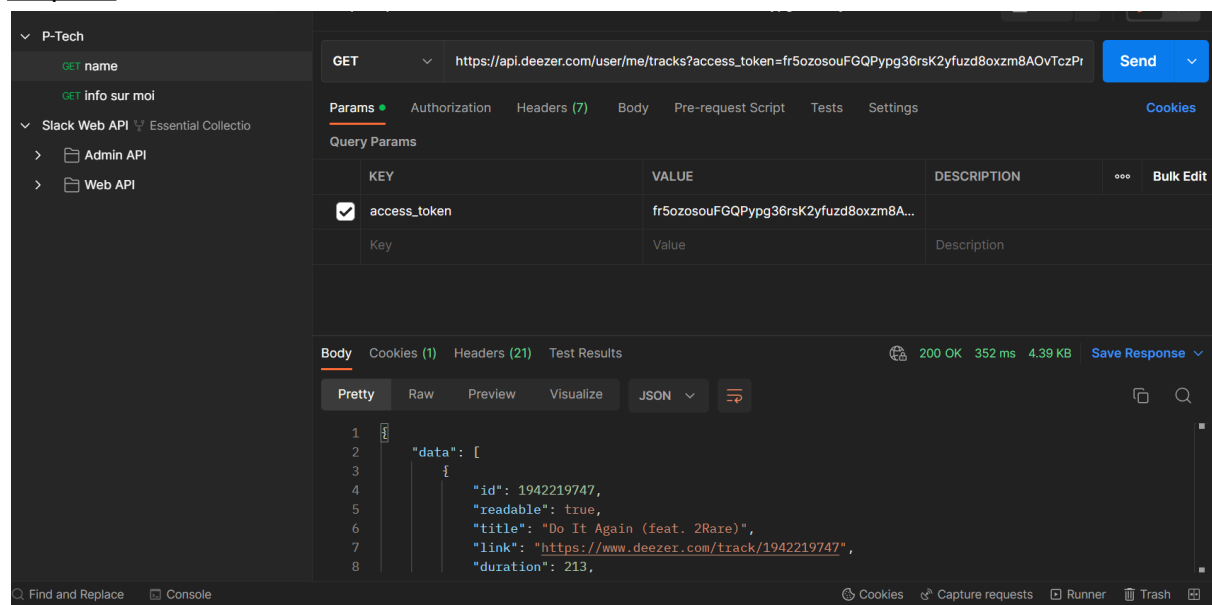
```

Find and Replace Console Cookies Capture requests Runner Trash

Etape 3:

```
C:\Users>cd..
C:\>cd C:\Users\your\Documents
C:\Users\your\Documents>cd C:\Users\your\Documents\node-deezer-main\examples
C:\Users\your\Documents\node-deezer-main\examples>cd with_express
C:\Users\your\Documents\node-deezer-main\examples\with_express>node app.js
Express server listening on port 3000
GET / 200 1206ms - 917b
(node:3552) [DEP0066] DeprecationWarning: OutgoingMessage.prototype._headers is deprecated
(Use `node --trace-deprecation ...` to show where the warning was created)
GET /stylesheets/style.css 200 59ms - 141b
{
  accessToken: 'frAJiyg9B5Q8bUY1GzynRWipHZobcpZwYrgDwKjRc4C7U8dmyPx',
  expires: 3600
}
GET /deezerCallback?code=fr774e982a618ebbd3c6c2ec07c71de2 302 341ms - 64b
Debug:: Checking Deezer token... {
  lifespan: 3600,
  token: 'frAJiyg9B5Q8bUY1GzynRWipHZobcpZwYrgDwKjRc4C7U8dmyPx',
  lastLogin: '2022-12-15T10:21:12.219Z'
}
GET /dashboard 200 109ms - 1.45kb
GET /stylesheets/style.css 304 9ms
```

Etape 4:



The screenshot shows a Postman interface with a GET request to `https://api.deezer.com/user/me/tracks?access_token=fr5ozosouFGQPyg36rsK2yfuzd8oxzm8A...`. The request is successful, returning a 200 OK status with a response time of 352 ms and a body size of 4.39 KB. The response body is formatted as JSON and shows a single track object.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> access_token	fr5ozosouFGQPyg36rsK2yfuzd8oxzm8A...	
Key	Value	Description

```
1  {
2    "data": [
3      {
4        "id": 1942219747,
5        "readable": true,
6        "title": "Do It Again (feat. 2Rare)",
7        "link": "https://www.deezer.com/track/1942219747",
8        "duration": 213,
```

Conclusion

Ce stage m'a permis de développer de nouvelles compétences dans le monde de l'informatique comme l'apprentissage du python du SQL ou bien des API que je ne connaissais pas auparavant avant cette période de stage.

Ce que je retiens de ce stage, c'est qu'il m'a permis de développer des habiletés déjà acquises et d'en apprendre de nouvelles.

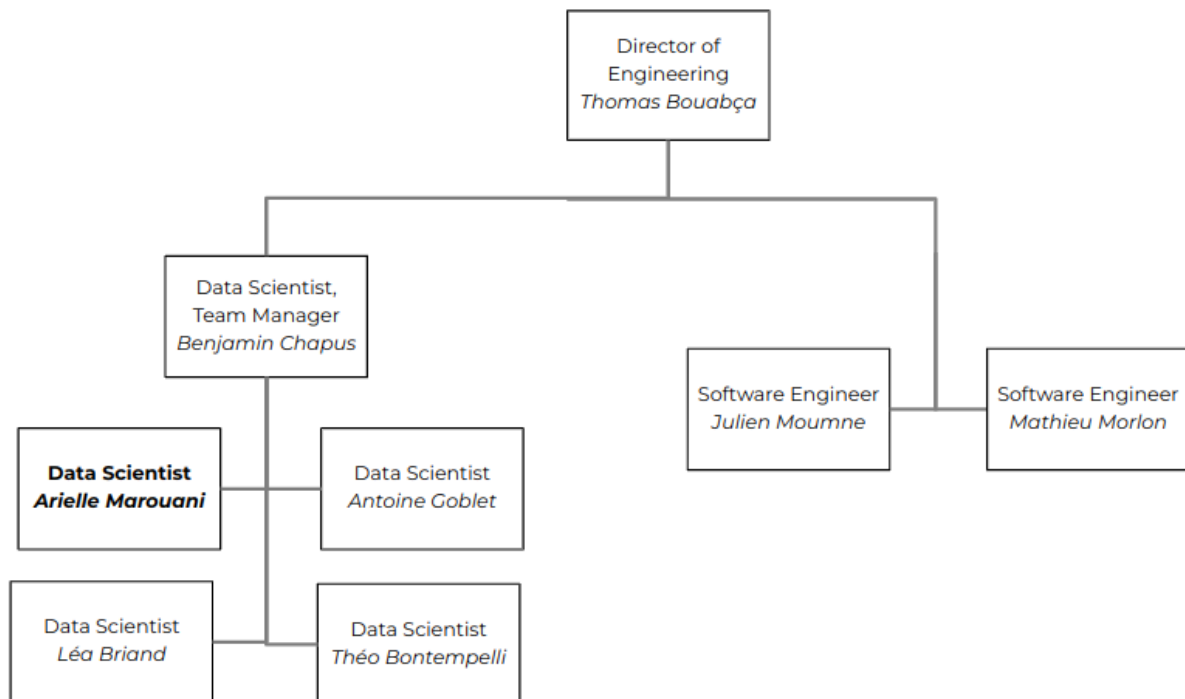
La manière de travailler dans l'entreprise est calme, il n'y a pas de stress ou de pression sur les employés, c'est une entreprise très ouverte d'esprit, il organise parfois des réunions pour proposer de nouveau projet sur l'amélioration de l'application.

Semaine 1 : Data et Product, langage SQL

La première semaine nous avons appris le langage SQL avec Arielle Marouani de l'équipe recommandation.

Présentation de l'équipe recommandation

L'équipe recommandation est constituée de six Data Scientists. Ils écrivent des algorithmes qui permettent de créer des fonctionnalités personnalisées pour les utilisateurs. Recommandation des chansons, sélection de playlists, proposition des artistes. Les performances des algorithmes sont ensuite analysées par des Data Analysts. Ils récoltent différentes "métriques" et en tirent des conclusions. Ces métriques peuvent être : Le nombre d'ajouts en playlist, le nombre de likes, le temps passé sur la playlist, le nombre de skips, le nombre d'albums par artistes.



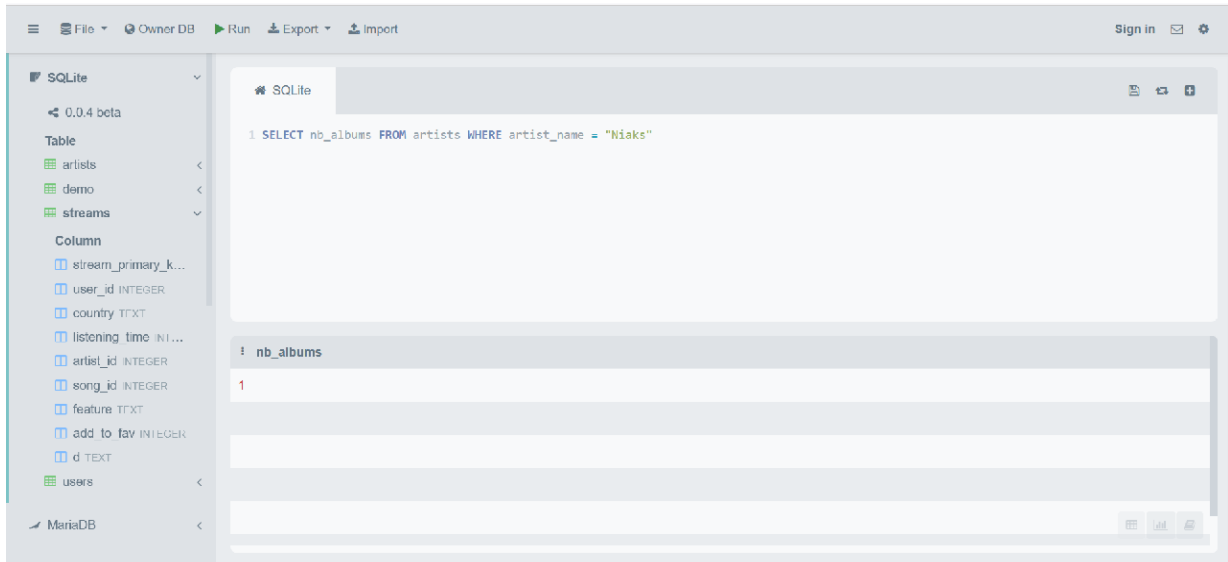
Observation

Pendant le stage, on a eu l'occasion d'assister à une réunion avec Arielle, membre de l'équipe recommandation qui était basée sur le langage du code SQL et son fonctionnement.

Tâches

Requêtes SQL

Écris une requête SQL qui donne le nombre d'albums de Niaks.



Analyse

On voit que Niaks possède 1 album. Pour permettre de réaliser ses TP nous avons utilisé une WebApp nommée SQLite et nous avons reçu par mail plusieurs documents contenant des informations sur des artistes nous devons les trier en fonction de la question.

Semaine 2 : Metadata curation

La deuxième semaine nous avons créé des playlists 100% deezer avec Deniz Mekik & Fouad Youssef de l'équipe Metadata curation.

Présentation de l'équipe

L'équipe Metadata est constituée de 5 Metadata Curation Deniz Mekik Leader des chargés de Metadata, Beranger Magni et Fouad Youssef sont en charge des Métadonnées.

La Metadata Curation est une perspective de développement rapide et massif de la culture data au sein d'une organisation. La constitution d'un plan de gestion et de curation des données permet tout d'abord de recenser les données produites.

- Améliore la qualité pour améliorer l'utilité des données
- Améliore grandement la chance de trouver les données
- Permet l'action des algorithmes de machine learning

Observation

Lors de cette semaine de stage nous avons eu la chance de créer des playlists 100% présentées par le compte officiel de Deezer. Pour les créer il suffisait de sélectionner à la main des titres un même artiste, créer une playlist avec le nom de l'artiste et les ajouter à cette playlist 100%. Les playlists 100% sont des playlists uniquement conçues avec des titres d'un artiste, les remix et ou les feat n'étaient pas acceptés sauf si l'artiste possède moins de 10 titres sans feat.

Tâches


Sélections des meilleurs titres

Le but était de sélectionner les meilleurs titres d'artistes dans la liste qui nous était proposée ses titres ne devaient pas comportaient de feat et de remix ces titres devaient également représenter l'artiste. Il nous était conseillé d'utiliser les titres les plus écoutés, notamment pour les mettre en avant dans la playlist.

Ajout des titres dans la playlist, Ajout des Covers.

Une fois les titres sélectionnés, on devait les ajouter à une playlist nommée 100% "le nom de l'artiste", les playlists devaient comporter en moyenne 20 titres et ne devaient pas dépasser les 30 titres. Une fois la playlist terminée, on devait y ajouter une image à la playlist, cette image nous était générée par l'équipe Metadata curation avec un outil interne de Deezer que seul l'entreprise peuvent y accéder,

100% Niaks





100% Niaks


Deezer Artist Editor

19 titres - 55 minutes - 71 fans - Mise à jour : il y a 3 semaines


À L'ÉCOUTE

TITRE	ARTISTE	ALBUM	AJOUTÉ		
 Hayla E	Niaks	Comparution imm...	28/11/2022	02:52	
 Ninetta E	Niaks	Comparution imm...	28/11/2022	03:32	

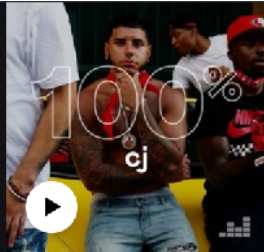
Playlist 100%



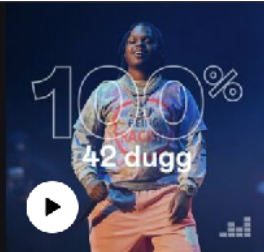
Créer une playlist




100% lil mabu
Publique




100% CJ
Publique




100% 42 Dugg
Publique




100% oby one



100% moha mmz



100% russ millions



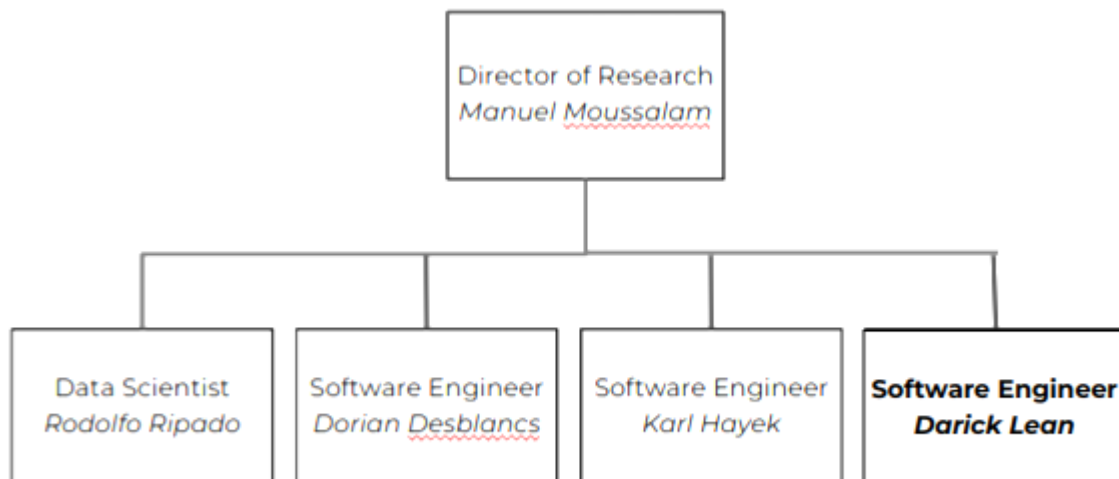
100% dusty locane

Semaine 3 : Data et Research, langage Python

La troisième semaine nous avons créé un Blind test en langage python avec Darick Lean de l'équipe Ingénieur logiciel (Software Engineer) dans l'équipe Recherche de la Division Innovation Metadata curation.

Présentation de l'équipe

Darick Lean (ingénieur logiciel), il analyse, conçoit, développe, teste, évalue et maintient des logiciels.



Observation

Nous avons créé un blind test au thème qu'on voulait qui pourraient être utilisés dans Deezer et nous avons reçu le code d'un jeu Blind Test en Python.

Tâches

Apprentissage du Python

Avant de commencer le blind test on devait s'entraîner sur plusieurs mini-jeux et cours sur un site nommé France IOI pour ensuite s'attaquer au BlindTest.

Création du Blind test

Tout d'abord nous avons reçu le code d'un blind test qu'on devait personnaliser à notre manière avec nos propres textes et images.

code du blind test en python

```

44     return levenshtein_ratio(true_answer, given_answer) > threshold
45
46
47 def BlindTestGame():
48     global vlc_player
49
50     label3 = tk.Label(root, text='', font=('helvetica', 30), fg='red')
51     canvas1.create_window(200, 210, window=label3)
52     x1 = entry1.get().lower()
53     ans = is_good_answer(artist_name.lower(), x1)
54
55     # vlc_player = vlc_player.copy()
56
57     if ans:
58         label3 = tk.Label(root, text='Bonne réponse', font=('helvetica', 30), fg='green')
59         canvas1.create_window(200, 210, window=label3)
60
61         vlc_player.stop()
62         get_random_question()
63         sleep(2)
64
65         vlc_player = vlc.MediaPlayer(f"tracks/{md5}.wav")
66         vlc_player.stop()
67         vlc_player.play()

```

Résultat test

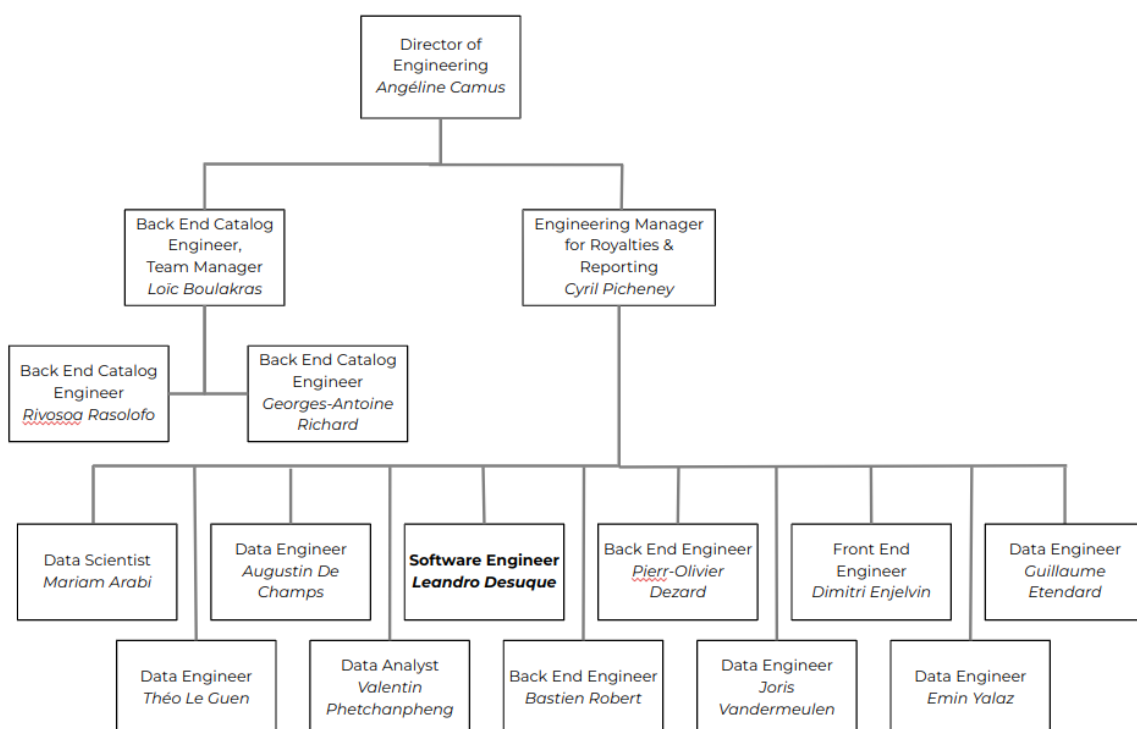


Semaine 4 : Royalties et Reporting, manipulation d'API

Lors de la dernière semaine nous avons appris à gérer des API (Application Programming Interface) avec Leandro Desuque de l'équipe ingénieur logiciel (Software Engineer) dans l'équipe Recherche de la Division Innovation.

Présentation de l'équipe

L'équipe Recherche de la division innovation est constituée de 8 Software Engineer. Leandro Desuque (ingénieur logiciel), il analyse, conçoit, développe, teste, évalue, maintient des logiciels et détecte les fraudes.



Observation

Nous avons créé une App Deezer pour développeurs et personnaliser des requêtes API (Application programming interface).

Tâches

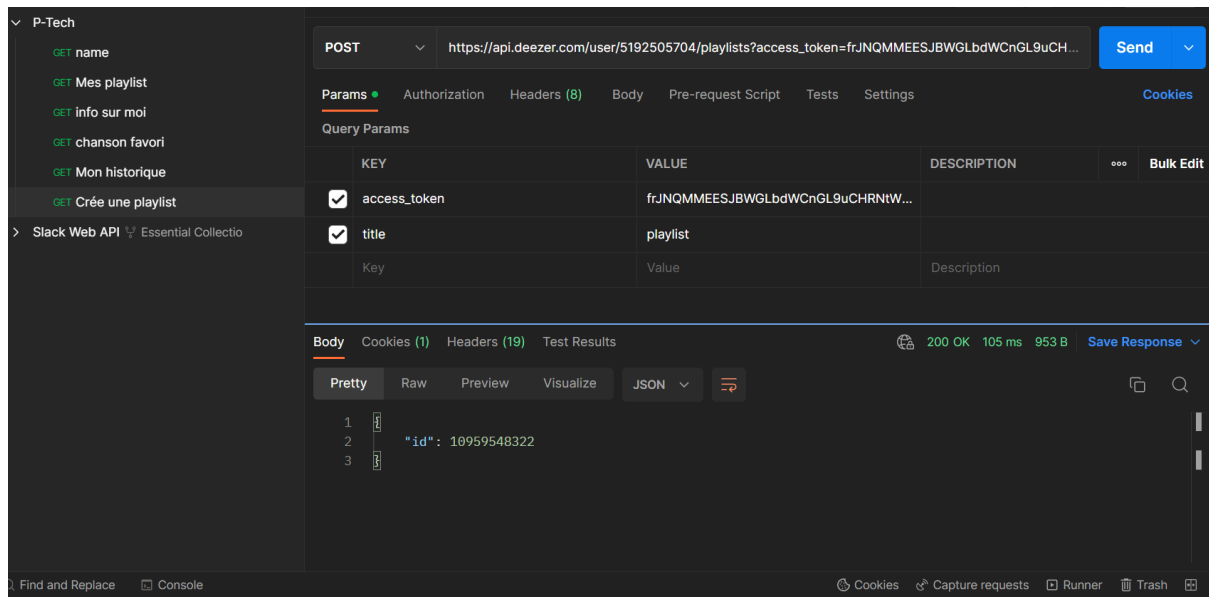
Création de l'App

Pour commencer nous nous sommes connectés au site pour développeurs de Deezer, ont a créé une App avec plein de paramètres qui nous a était donné et nous l'avons mis en local, récupérer le token du compte connecté à l'app ensuite nous étions en mesure d'utiliser l'API et de personnaliser les commandes URL selon la tâche qu'il fallait faire.

Utilisation de l'API

Après l'app installée on a personnalisé les commandes URL pour recevoir différentes informations comme supprimer, modifier, ajouter, personnaliser un titre ou une playlist ou aussi savoir l'historique de l'utilisateur.

Créer une playlist



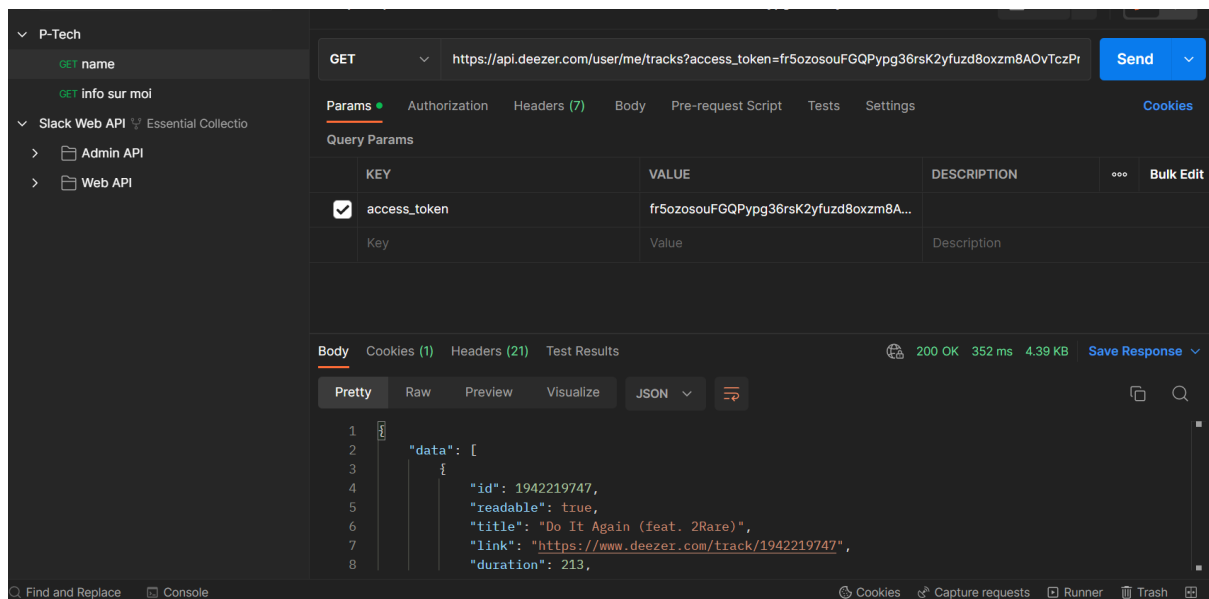
The screenshot shows a Postman interface for a POST request to the URL `https://api.deezer.com/user/5192505704/playlists?access_token=frJNQMMEEsJBWGLbdWCnGL9uCH...`. The request is configured with the following parameters:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> access_token	frJNQMMEEsJBWGLbdWCnGL9uCH...	
<input checked="" type="checkbox"/> title	playlist	

The response body is shown in JSON format:

```
1 {
2   "id": 10959548322
3 }
```

Chanson favori



The screenshot shows a Postman interface for a GET request to the URL `https://api.deezer.com/user/me/tracks?access_token=fr5ozosouFGQPypg36rsK2yfuzd8oxzm8A...`. The request is configured with the following parameters:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> access_token	fr5ozosouFGQPypg36rsK2yfuzd8oxzm8A...	

The response body is shown in JSON format:

```
1 {
2   "data": [
3     {
4       "id": 1942219747,
5       "readable": true,
6       "title": "Do It Again (feat. 2Rare)",
7       "link": "https://www.deezer.com/track/1942219747",
8       "duration": 213,

```