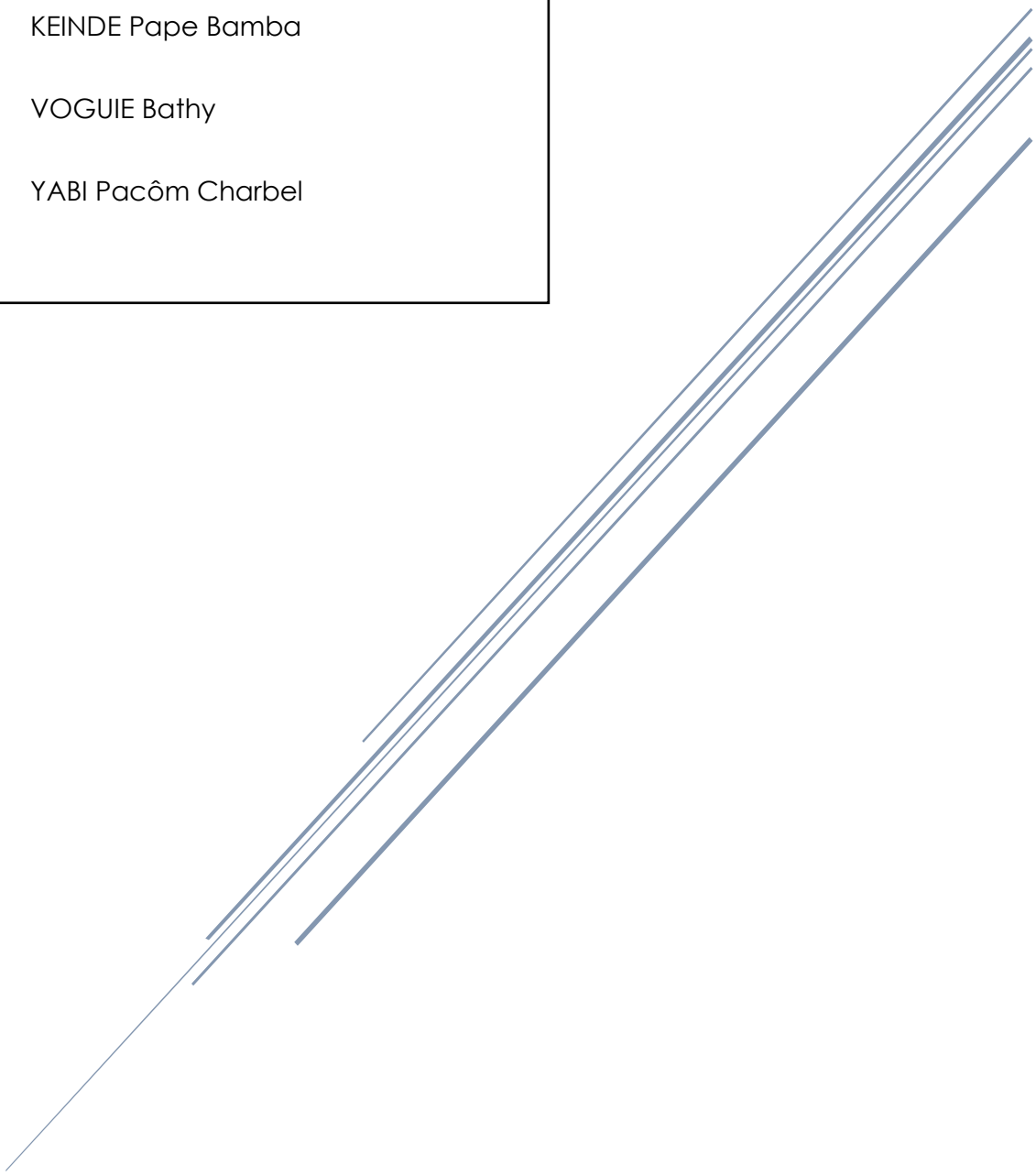
SERVICE DE STREAMING MUSICAL – SelfHits

- DOGHA N'KOUEKEM Kevin Delano
- KEINDE Pape Bamba
- VOGUIE Bathy
- YABI Pacôm Charbel



Licence Professionnelle – MDS 2023

NoSQL

CONTEXTE

Il y a eu le vinyle, le CD, la musique en téléchargement, et maintenant la musique en streaming. Depuis 7 ans, le marché de la musique est reparti à la hausse, en exemple avec une augmentation de 5,4% en 2016, 3,9% en 2017 et 1,8% en 2018. Pour **un chiffre d'affaires de 735 millions d'euros**. Et c'est d'abord grâce à la musique en streaming (Source : https://www.ariase.com/mobile/dossiers/streaming-musical). Aujourd'hui en France, un consommateur sur deux utilise un service de streaming musical. C'est alors que nous intervenons.

En effet, nous sommes une startup nommée "**SelfHits**" et nous souhaitons réaliser une base de données MongoDB en service de streaming musical permettant de trouver n'importe quelle musique d'artistes dans des catégories variées (Pop, Rap, R'N'B…). Il sera également possible de se faire une playlist de nos chansons favorites, voir les titres les plus lus, obtenir des informations sur un(e) artiste, ses albums produits.

DESIGN DU PROJET

- ### Business case

SelfHits sera un service numérique qui proposera de la musique et qui offrira à ses utilisateurs un accès à des millions de titres de tout genre et d'auteurs des quatre coins du monde. Ses fonctionnalités de base seront totalement gratuites (écouter de la musique, lyrics, création de playlist...), mais il y aura une version premium (**SelfHits Premium**) qui elle sera payante. SelfHits sera disponible sur plusieurs appareils notamment ordinateurs, téléphones, tablettes, télévisions ou encore dans les voitures et juste à partir de ses identifiants l'utilisateur pourra basculer d'un appareil à l'autre.

L'équipe qui sera engagée dans la réalisation de ce projet est constituée de :

- Bathy VOGUIE (Administrateur de bases de données),
- Pacôm Charbel YABI (Software Architect & DevOp),
- Pape Bamba KEINDE (Chef de projet)
- Kevin Delano DOGHA NKOUEKEM (Développeur Web).

Outre l'équipe porteuse du projet, nous envisageons une collaboration avec des prestataires externes en l'occurrence deux développeurs Web et un testeur responsable de la qualité de code produit.

La durée totale du projet est estimée à 11 mois ; son coût est estimé à 20 k€ en compte tenu des ressources humaines externes, la durée des tâches, la durée totale du projet et les ressources matérielles nécessaires pour la mise en œuvre de l'application.

- **Estimation de la taille de données (disk space)**

La taille de document maximale des documents dans une base de données MongoDB permet de s'assurer qu'un seul document ne puisse utiliser une quantité excessive de RAM ou, pendant la transmission, une quantité excessive de bande passante. Pour stocker des documents plus grands que la taille maximale, MongoDB fournit l'API GridFS.

Dans notre cas, nous organisons la sauvegarde des documents de manière à ne pas excéder la taille maximale de 15 Mo, soit 1 Mo de moins que la taille maximale fixée pour un document de collection MongoDB.

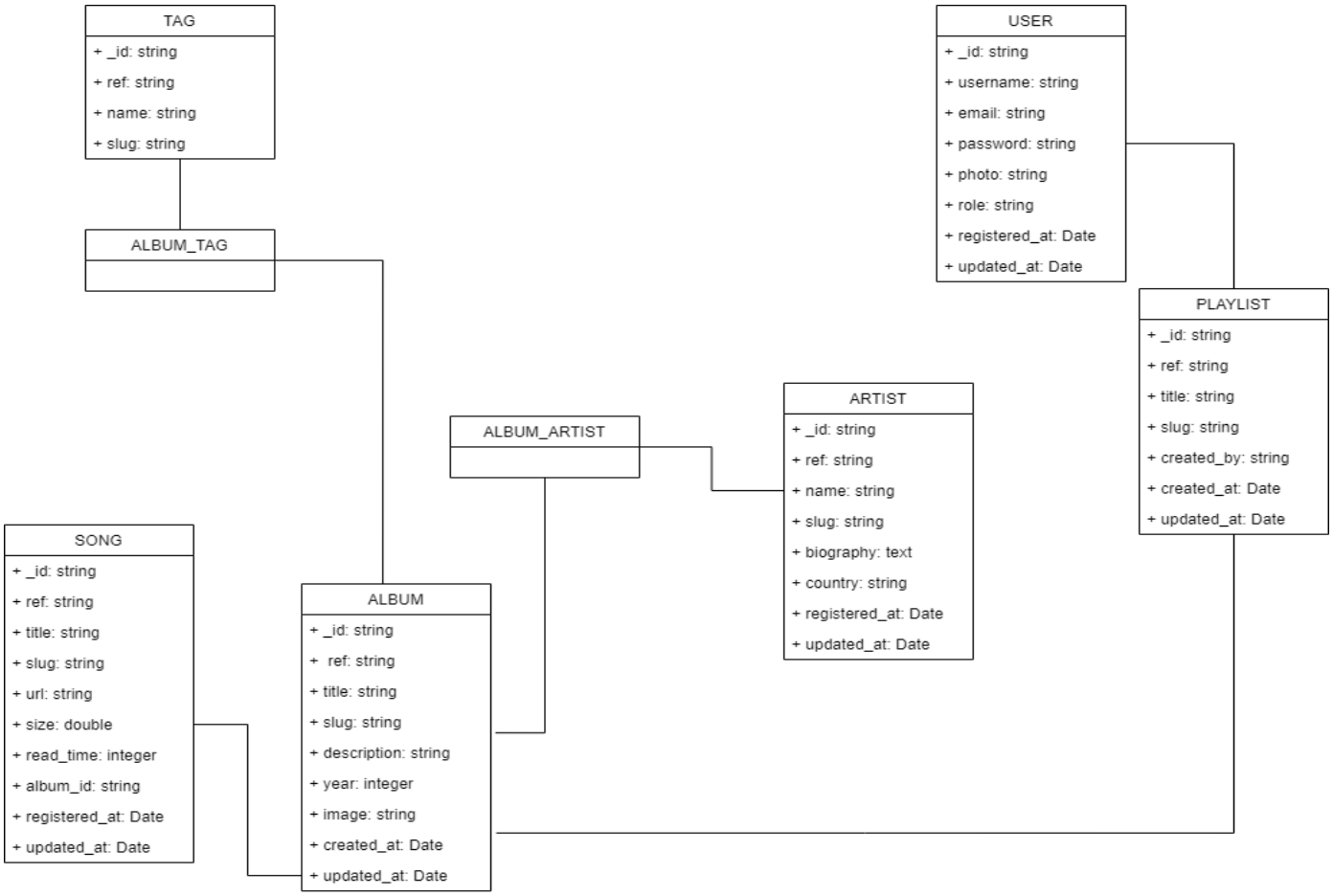
Ainsi donc, nous estimons pour les **6 premiers mois d'exploitation** de la plateforme, pouvoir compter près de **100 000 albums accessibles** pour près de **1 000 000** utilisateurs. En supposant que chaque album puisse compter **10 titres** pesant chacun **3Mo** en moyenne, nous estimons l'espace du disque pour l'hébergement nécessaire à **2,86 To**. Soit environ **3 To** d'espace mémoire.

- ## Diagramme UML avec les relations entre les collections

Le diagramme ci-dessous permet d'exprimer visuellement les besoins de notre plateforme. Il décrit de ce fait la structure des collections de notre base de données MongoDB tout en visualisant les différents types d'objets au sein du système et les types de relations statiques qui existent entre eux. Il illustre également les attributs des classes. Celui-ci présente 6 collections que sont :

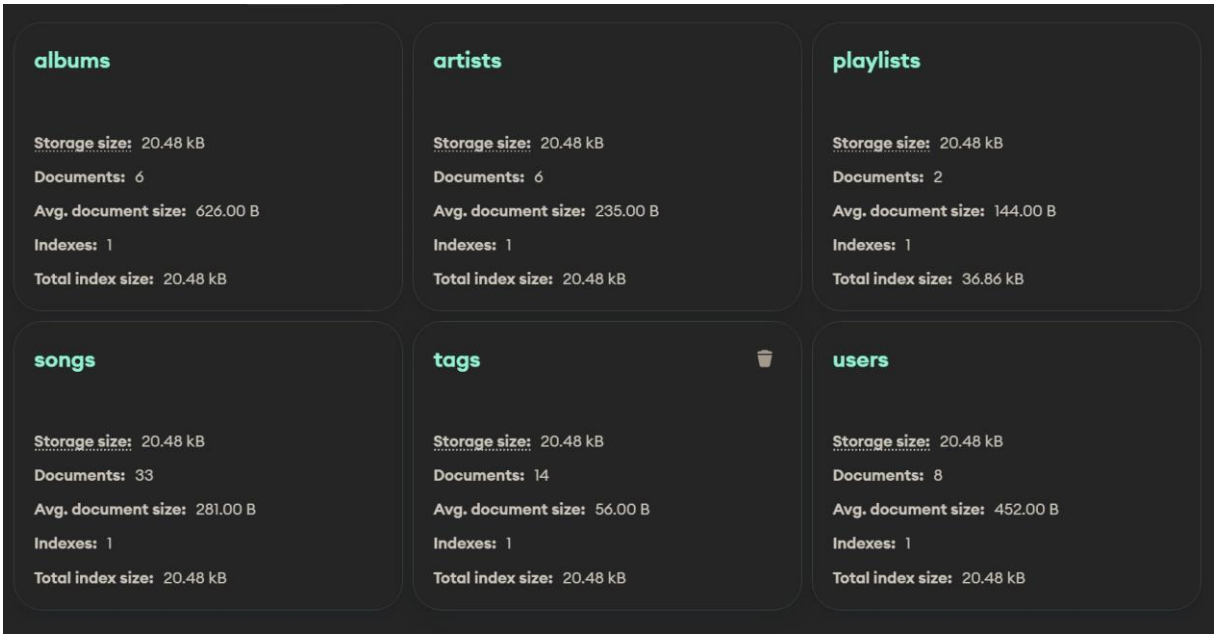
- **TAG** : il s'agit de la collection chargée de stocker les différentes catégories de musiques disponibles sur la plateforme.
- **USER** : elle stocke la liste des utilisateurs enregistrées sur la plateforme et permet un accès direct à leur type de compte (définissant les actions qui leur est possible de réaliser).
- **ARTIST** : elle stocke les informations sur les artistes dont les titres & albums sont disponible en streaming.
- **ALBUM** : elle stocke la liste des albums disponibles
- **SONG** : elle stocke les différents titres qui composent les albums. Il s'agit également de la collection la plus importante de la base de données, représentant la collection la plus requêtée en lecture.
- **PLAYLIST** : elle permet de stocker les playlists d'utilisateurs. Une playlist représente une collection d'albums étiquetées par un utilisateur organisée selon ses préférences.

Remarque : Nous ajoutons dans chaque collection une référence permettant de faire la liaison avec d'autres collections, l'objectif étant un gain de temps dans l'exécution des requêtes.


TAG
+ _id: string
+ ref: string
+ name: string
+ slug: string
ALBUM_TAG
USER
+ _id: string
+ username: string
+ email: string
+ password: string
+ photo: string
+ role: string
+ registered_at: Date
+ updated_at: Date
PLAYLIST
+ _id: string
+ ref: string
+ title: string
+ slug: string
+ created_by: string
+ created_at: Date
+ updated_at: Date
ALBUM_ARTIST
ARTIST
+ _id: string
+ ref: string
+ name: string
+ slug: string
+ biography: text
+ country: string
+ registered_at: Date
+ updated_at: Date
SONG
+ _id: string
+ ref: string
+ title: string
+ slug: string
+ url: string
+ size: double
+ read_time: integer
+ album_id: string
+ registered_at: Date
+ updated_at: Date
ALBUM
+ _id: string
+ ref: string
+ title: string
+ slug: string
+ description: string
+ year: integer
+ image: string
+ created_at: Date
+ updated_at: Date

IMPLÉMENTATION DE LA BASE DE DONNÉS

- **Structure de la base de données : liste des collections**



- **Quelques requêtes et quantification** [i]

  - **Requête d'écriture étoile :** Insertion de nouveaux titres musicaux

```
> db.songs.insert({
      "title": "New independant single",
      "url": "https://www.rireetchansons.fr/podcasts/lappel-trop-con/augmentation-fromagere-l-appel-trop-con-de-rire-chansons-1",
      "duration": "3 min",
      "size": "3.2",
      "read_time": "15",
      "registerd_at": new Date(),
      "registerd_by": ""
    })
< 'DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.'
< {
    acknowledged: true,
    insertedIds: {
      '0': ObjectId("6421d6d01c8c9003e2c539f9")
    }
  }
SelfHits>
```

- **Requête de lecture étoile :** Recherche d'albums et/ou de titres spécifiques

```
> db.songs.find().limit(10)
< {
    _id: ObjectId("64074b511f567840c6d014a7"),
    title: 'New independant single',
    url: 'https://www.rireetchansons.fr/podcasts/lappel-trop-con/augmentation-fromagere-l-appel-trop-con-de-rire-chansons-1',
    duration: '3 min',
    size: '3.2',
    read_time: '15',
    registerd_at: 2023-03-07T14:33:53.584Z,
    registerd_by: ''
  }
  {
    _id: ObjectId("64074b511f567840c6d014a8"),
    title: 'New independant single',
    url: 'https://www.rireetchansons.fr/podcasts/lappel-trop-con/augmentation-fromagere-l-appel-trop-con-de-rire-chansons-1',
    duration: '3 min',
    size: '3.2',
    read_time: '15',
    registerd_at: 2023-03-07T14:33:53.584Z,
    registerd_by: ''
  }
  {
    _id: ObjectId("64074b511f567840c6d014a9"),
    title: 'New independant single',
    url: 'https://www.rireetchansons.fr/podcasts/lappel-trop-con/augmentation-fromagere-l-appel-trop-con-de-rire-chansons-1',
    duration: '3 min',
```

- Nombre de titres par albums

```
> db.songs.aggregate([
    {
      $group: {
        _id: '$album_id',
        count: { $sum: 1 }
      }
    }
  ])
< {
    _id: null,
    count: 4
  }
  {
    _id: ObjectId("6406caff1f567840c6d01455"),
    count: 6
  }
  {
    _id: ObjectId("6406caff1f567840c6d01457"),
    count: 6
  }
```

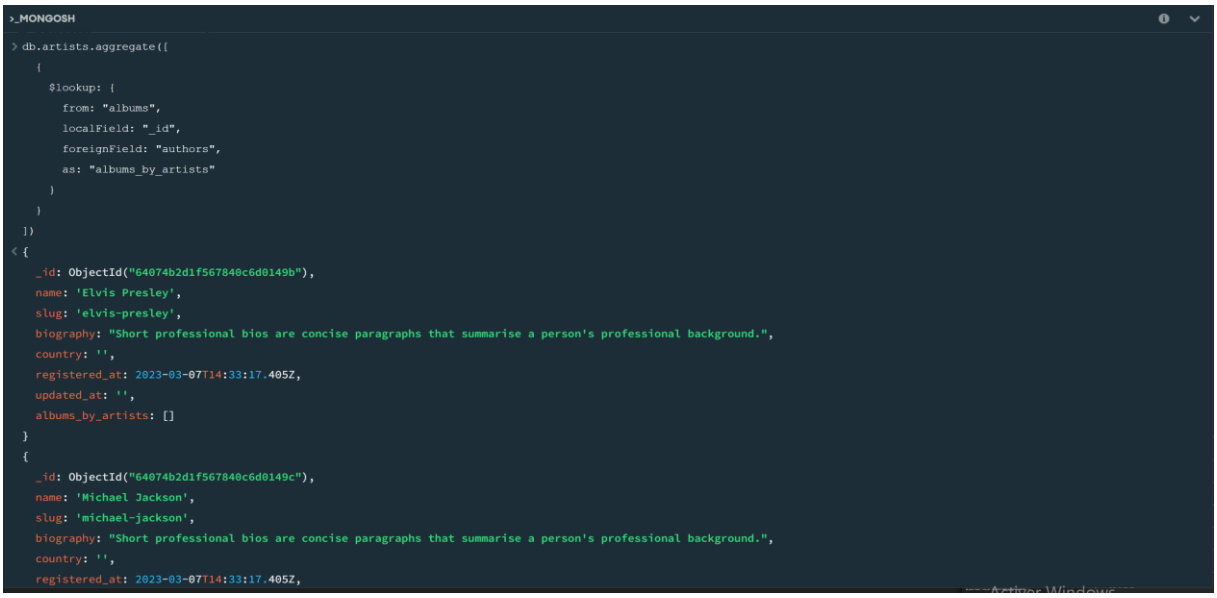
- Liste des titres indépendants

```
> db.songs.find({ album_id: null }).projection({ title: 1 })
< {
    _id: ObjectId("64074b511f567840c6d014a7"),
    title: 'New independant single'
  }
  {
    _id: ObjectId("64074b511f567840c6d014a8"),
    title: 'New independant single'
  }
  {
    _id: ObjectId("64074b511f567840c6d014a9"),
    title: 'New independant single'
  }
  {
    _id: ObjectId("6421d6d01c8c9003e2c539f9"),
    title: 'New independant single'
  }
```

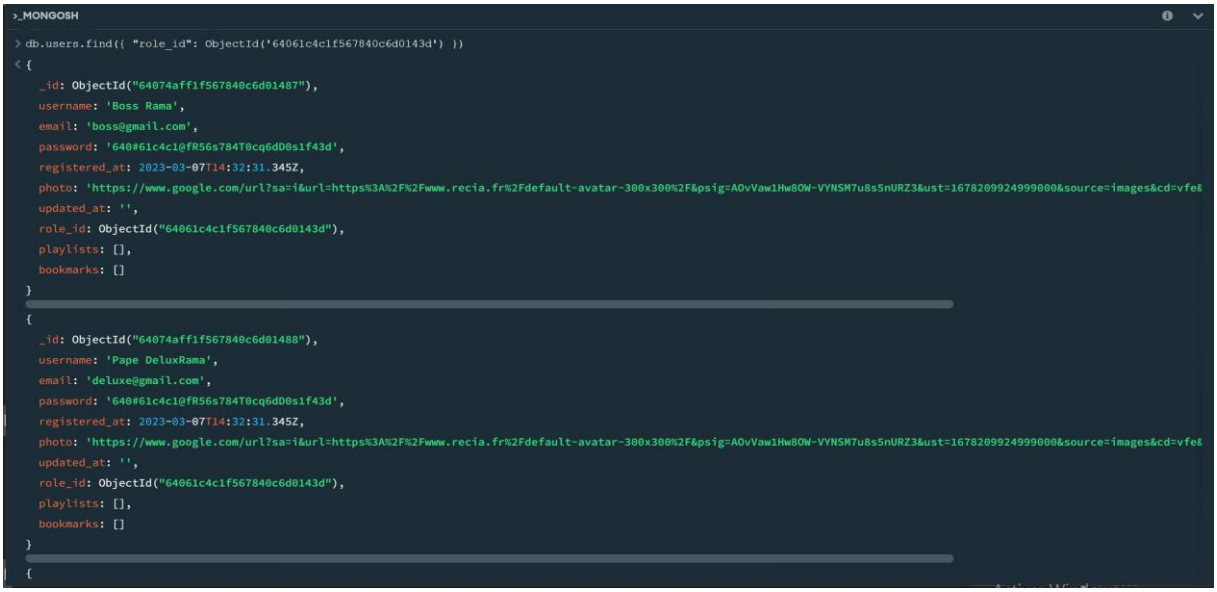
- Liste des albums

```
> db.albums.find()
< {
    _id: ObjectId("6421fcbd1c8c9003e2c539fa"),
    title: 'A New Day : Live in Las Vegas\t',
    slug: 'a-new-day-live-in-las-vegas',
    description: 'Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat eaque accusantium magnam quae dolor sit sunt blanditiis, repellendus pariatur
    year: '2004',
    image: 'https://www.francebleu.fr/s3/cruiser-production/2017/07/0c3b29c2-d074-4359-b6bf-e90242b9b654/1200x680_un_peu_de_nous1.jpg',
    authors: [
      ObjectId("640624601f567840c6d01451")
    ],
    tags: [
      {
        ref: 2
      },
      {
        ref: 4
      },
      {
        ref: 7
      }
    ],
    created_at: 2023-03-27T20:29:49.844Z,
    updated_at: ''
  }
  {
    _id: ObjectId("6421fcbd1c8c9003e2c539fb"),
    title: 'Au cœur du Stade',
    slug: 'au-cœur-du-stade',
```

- Liste des artistes avec leurs albums



- Liste des abonnés (sans les administrateurs)



- Création de playlist pour un utilisateur

```
>_MONGOSH
> db.playlists.insertOne({
    "name": "Mes Favoris Rock",
    "slug": "mes-favoris-rock",
    "user_id": ObjectId('640622721f567840c6d01446'),
    "albums": [
      {ref: 2},
      {ref: 4},
      {ref: 7},
    ]
  })
< {
    acknowledged: true,
    insertedId: ObjectId("6429f66ee837a426dfed2a3c")
  }
SelfHits>
```

- Mise à jour de playlist dont l'id vaut : **64074d9b1f567840c6d014c8**

```
>_MONGOSH
> db.playlists.updateOne({
    "_id": ObjectId('64074d9b1f567840c6d014c8')
    },
    {
      $push :{ "albums": {ref: 7} }
    })
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0
  }
SelfHits>
```

- Liste des playlists de l'utilisateur : **640622721f567840c6d01446**

```
>_MONGOSH

> db.playlists.find({ "user_id": ObjectId('640622721f567840c6d01446') }).pretty()
< {
    _id: ObjectId("6429f650e837a426dfed2a3b"),
    name: 'Mes Favoris Rock',
    slug: 'mes-favoris-rock',
    user_id: ObjectId("640622721f567840c6d01446"),
    albums: [
      {
        ref: 2
      },
      {
        ref: 4
      },
      {
        ref: 7
      }
    ]
  }
  {
    _id: ObjectId("6429f66ee837a426dfed2a3c"),
    name: 'Mes Favoris Rock',
    slug: 'mes-favoris-rock',
    user_id: ObjectId("640622721f567840c6d01446"),
    albums: [
      {
        ref: 2
      },
```

- Liste des albums datant de 2004 (par exemple)

```
>_MONGOSH

> db.albums.find({ "year": "2004" })
< {
    _id: ObjectId("642206bc1c8c9003e2c53a0c"),
    title: 'A New Day : Live in Las Vegas\t',
    slug: 'a-new-day-live-in-las-vegas',
    ref: 1,
    description: 'Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat eaque accusantium magnam quae dolor sit sunt blanditiis,
    year: '2004',
    image: 'https://www.francebleu.fr/s3/cruiser-production/2017/07/0c3b29c2-d074-4359-b6bf-e90242b9b654/1200x680_un_peu_de_nous1.jpg',
    authors: [
      {
        ref: 1
      },
      {
        ref: 2
      }
    ],
    tags: [
      {
        ref: 2
      },
      {
        ref: 4
      },
      {
        ref: 7
      }
```

- Top N des titres les plus écoutés

```
>_MONGOSH
> db.songs.find().projection({"_id": 0, "title": 1, "read_time": 1}).sort({ "read_time": -1}).limit(10)
< {
    title: 'New single',
    read_time: 807
  }
  {
    title: 'New single',
    read_time: 502
  }
  {
    title: 'New single',
    read_time: 415
  }
  {
    title: 'New single',
    read_time: 201
  }
  {
    title: 'New single',
    read_time: 121
  }
  {
    title: 'New single',
    read_time: 105
  }
  {
    title: 'New single',
    read_time: 65
  }
  {
    title: 'New single',
    read_time: 25
```

[i] Retrouvez ci-jointe, un dépôt GitHub avec la liste de toutes les requêtes ci-dessus énumérées.