

Conception de la base de données MongoDB :

Voici les données comment ils ont été au début :

1. ****workers**** : **Stocke** les informations des travailleurs de l'**industrie** (réalisateurs et acteurs)
 - ``id`` : **Identifiant** unique (UUID)
 - ``name`` : **Nom** du travailleur
2. ****genres**** : **Catégorise** les différents genres de films
 - ``id`` : **Identifiant** unique (UUID)
 - ``name`` : **Nom** du genre (unique)
3. ****awards**** : **Répertorie** les différents prix
 - ``id`` : **Identifiant** unique (UUID)
 - ``name`` : **Nom** du prix
 - ``category`` : **Catégorie** du prix
4. ****movies**** : **Contient** les informations sur les films
 - ``id`` : **Identifiant** unique (UUID)
 - ``title`` : **Titre** du film
 - ``year`` : **Année** de sortie
 - ``director_id`` : **Référence** au réalisateur (lié à la table workers)
 - ``metadata`` : **Données** supplémentaires au format JSONB
5. ****movies_genres**** : **Table** de jonction entre films et genres (relation many-to-many)
 - ``movie_id`` : **Référence** au film
 - ``genre_id`` : **Référence** au genre
6. ****movie_actors**** : **Associe** les acteurs à leurs films et rôles (relation many-to-many)
 - ``id`` : **Identifiant** unique (UUID)
 - ``movie_id`` : **Référence** au film
 - ``actor_id`` : **Référence** à l'**acteur** (lié à la table workers)
 - ``role`` : **Nom** du rôle joué
7. ****movie_awards**** : **Associe** les prix aux films (relation many-to-many)
 - ``movie_id`` : **Référence** au film
 - ``award_id`` : **Référence** au prix
 - ``year`` : **Année** de remise du prix
8. ****users**** : **Stocke** les informations des utilisateurs de l'**application**
 - ``id`` : **Identifiant** unique (UUID)
 - ``name`` : **Nom** de l'**utilisateur**
9. ****favorite_genres**** : **Associe** les genres préférés aux

```

utilisateurs (relation many-to-many)
- `user_id` : Référence à l'utilisateur
- `genre_id` : Référence au genre

10. **watch_history** : Enregistre l'historique de visionnage des
utilisateurs (relation many-to-many)
- `id` : Identifiant unique (UUID)
- `user_id` : Référence à l'utilisateur
- `movie_id` : Référence au film
- `watched_on` : Date de visionnage

11. **ratings** : Stocke les évaluations et critiques des
utilisateurs
- `id` : Identifiant unique (UUID)
- `user_id` : Référence à l'utilisateur
- `movie_id` : Référence au film
- `rating` : Note attribuée (entre 0 et 5)
- `review` : Texte de la critique

```

Ce qu'on doit faire c'est faire une modélisation mongodb de ces données , après on passe a la creation et au peuplement des de données Et finalement on fait des requêtes pour l'analyse de performance

La base de données MongoDB a été conçue pour répondre efficacement aux besoins d'une plateforme de streaming, notamment :

- Afficher les fiches complètes des films
- Calculer et afficher les notes moyennes des films
- Faire des recommandations personnalisées basées sur les acteurs

Pour cela, on utilise un modèle documentaire dénormalisé , typique de MongoDB, qui permet de réduire les jointures et d'améliorer les performances en lecture .

Pour movies : (Centrale contient les infos principale)

```

{
  "_id": "uuid-movie-1",
  "title": "Inception",
  "year": 2010,
  "director": {
    "_id": "uuid-worker-1",
    "name": "Christopher Nolan"
  }
}

```

```

},
"genres": [
  { "_id": "uuid-genre-1", "name": "Action" },
  { "_id": "uuid-genre-2", "name": "Sci-Fi" }
],
"actors": [
  {
    "_id": "uuid-worker-2",
    "name": "Leonardo DiCaprio",
    "role": "Dominick Cobb"
  }
],
"awards": [
  {
    "_id": "uuid-award-1",
    "name": "Oscar",
    "category": "Best Cinematography",
    "year": 2011
  }
],
"metadata": {
  "runtime": 148,
  "language": "English",
  "budget": 160000000
},
"average_rating": 4.7,
"ratings_count": 1500
}

```

Pour Utilisateur : (pour les préférences et historiques)

```

{
  "_id": "uuid-user-1",
  "name": "Alice Dupont",
  "favorite_genres": [
    { "_id": "uuid-genre-1", "name": "Action" },
    { "_id": "uuid-genre-3", "name": "Comedy" }
  ]
}

```

Pour watch_history : (pour les recommandations basées sur visionnage)

```

{
  "_id": "uuid-watch-1",
  "user_id": "uuid-user-1",
  "movie_id": "uuid-movie-1",
  "movie_title": "Inception",
  "watched_on": "2023-05-20T00:00:00Z"
}

```

Pour workers : (Pour les relations awteur/film)

```
{  
  "_id": "uuid-worker-1",  
  "name": "Christopher Nolan"  
}
```