

Licence 3 Informatique – Université de Nantes - UFR Sciences et Techniques

Devoir Base de Données 2

Étudiants : Garnier Cyprien – Benbrik–Kerhamon Ismaël – Dupont Romain –
Leguillier Hugo – Groupe 684J

Février 2021



UNIVERSITÉ DE NANTES

Sommaire

1	Introduction	3
2	Déroulement du processus et distribution des tâches:	3
2.1	La table	3
2.2	Tuples	4
2.3	Les clés	4
2.4	Les dépendances fonctionnelles	4
2.5	Algorithmes de normalisation	5
2.5.1	Algorithme de synthèse	5
2.5.2	Algorithme de décomposition	7
2.5.3	Choix du modèle	7
3	Modèle Relationnel de votre base de données (tables, clés, types, cardinalités, etc).	8
4	Explication des fonctions, triggers, vues, etc. de votre base de données.	9
5	Atouts, améliorations possibles et critiques de votre travail.	9
6	Conclusion.	9

1 Introduction

Durant ces temps de confinement et malgré notre statut d'étudiant, nous ne le cachons pas, nous sommes fans de Netflix. Le but du travail étant de faire une base de données, comment aurions nous pu passer à côté de cette opportunité ? Dans la suite de ce travail, nous allons essayer de faire une base de données la plus fidèle possible, alliant réalisme et performance, en ayant en tête les réelles complications des entreprises telles que Netflix.

2 Déroulement du processus et distribution des tâches:

2.1 La table

Premièrement, nous avons créé notre table ainsi que nos attributs. Ce travail a été réalisé en groupe entier, chacun proposant ses idées/ses avis. Durant cette partie, nous nous sommes grandement inspiré du site internet Netflix.

Au final, nous avons une table composée de 21 attributs, définis ci-dessous :

- `id_Oeuvre`, représentant un identifiant pour chaque oeuvre, une oeuvre étant un film ou une série. Deux épisodes d'une même série auront le même `id_Oeuvre`.
- `num_saison`, représentant un identifiant pour chaque saison d'une série.
- `num_episode`, représentant un identifiant pour chaque épisode d'une saison d'une série.
- `nom_Episode`, représentant le nom d'un épisode d'une série.
- `date_sortie`, représentant l'année de sortie d'une oeuvre.
- `genres`, représentant le(s) genre(s) de chaque oeuvre.
- `distribution`, représentant la distribution de l'oeuvre.
- `adjectif`, résume en quelques adjectifs une oeuvre.
- `description`, fait un résumé plus poussé que adjectif de l'oeuvre.
- `categorie_age`, représente l'âge minimum conseillé.
- `profil`, représente le nom de profil d'un `id_sousCompte`.
- `like`, peut prendre les valeurs ("oui" ou "non" ou "non choisi") et représente si un profil d'un compte à ou non like l'oeuvre.
- `liste`, peut prendre les valeurs ("oui" ou "non") et représente si l'oeuvre est, ou non dans la liste d'un profil d'un compte
- `abonnement`, définit le type d'abonnement du compte. Peut prendre les valeurs ("Essentiel" ou "Standard" ou "Premium").
- `id_sousCompte`, identifiant d'un sousCompte d'un compte
- `mail`, représentant le mail associée à un compte
- `telephone`, représentant le numéro de téléphone associé à un compte
- `mot_de_passe`, représentant le mot de passe associé à un compte
- `temps_visionnage`, représentant la minute à laquelle un profil d'un compte s'est arrêté sur une oeuvre.

2.2 Tuples

Cliquez sur ce lien pour voir notre base avec ses tuples: [Nos Tuples](#)

2.3 Les clés

Gauche	Millieu	Droit
id_oeuvre	num_episode	nom_oeuvre
num_saison	nom_episode	duree
id_sousCompte	mail	date_sortie
mail	telephone	genre
		distribution
		adjectif
		description
		categorie_age
		profil
		liste
		like
		abonnement
		mdp
		temps_visionnage

2.4 Les dépendances fonctionnelles

DF(1) id_Oeuvre \rightarrow nom_Oeuvre.

DF(2) id_Oeuvre, num_saison, nom_episode \rightarrow num_episode, durée, date_sortie, distribution, categorie_age, description, genres, adjectif

DF(3) id_Oeuvre, num_saison, num_episode \rightarrow durée, nom_episode, date_sortie, distribution, categorie_age, description, genres, adjectif

DF(4) mail \rightarrow abonnement, téléphone, mot de passe

DF(5) téléphone \rightarrow abonnement, mail, mot de passe

DF(6) téléphone, id_sousCompte \rightarrow profil, liste, like

DF(7) mail, id_sousCompte \rightarrow profil, liste, like

DF(8) téléphone, id_sousCompte, id_Oeuvre, num_saison, num_episode \rightarrow temps_visionnage

DF(9) mail, id_sousCompte, id_Oeuvre, num_saison, num_episode \rightarrow temps_visionnage

DF(10) téléphone, id_sousCompte, id_Oeuvre, num_saison, nom_episode \rightarrow temps_visionnage

DF(11) mail, id_sousCompte, id_Oeuvre, num_saison, nom_episode \rightarrow temps_visionnage

2.5 Algorithmes de normalisation

2.5.1 Algorithme de synthèse

1. Calcul des CV(DF)

Après avoir créé nos dépendances fonctionnelles, nous pouvons maintenant les mettre sous forme de couverture minimale.

Nous obtenons la liste suivante :

- 1) id_Oeuvre \rightarrow nom_oeuvre
- 2) id_Oeuvre, num_saison, nom_episode \rightarrow num_episode
- 3) id_Oeuvre, num_saison, nom_episode \rightarrow durée
- 4) id_Oeuvre, num_saison, nom_episode \rightarrow date_sortie
- 5) id_Oeuvre, num_saison, nom_episode \rightarrow distribution
- 6) id_Oeuvre, num_saison, nom_episode \rightarrow categorie_age
- 7) id_Oeuvre, num_saison, nom_episode \rightarrow description
- 8) id_Oeuvre, num_saison, nom_episode \rightarrow genres
- 9) id_Oeuvre, num_saison, nom_episode \rightarrow adjectif
- 10) id_Oeuvre, num_saison, num_episode \rightarrow nom_episode
- 11) id_Oeuvre, num_saison, num_episode \rightarrow durée
- 12) id_Oeuvre, num_saison, num_episode \rightarrow date_sortie
- 13) id_Oeuvre, num_saison, num_episode \rightarrow distribution
- 14) id_Oeuvre, num_saison, num_episode \rightarrow categorie_age
- 15) id_Oeuvre, num_saison, num_episode \rightarrow description
- 16) id_Oeuvre, num_saison, num_episode \rightarrow genres
- 17) id_Oeuvre, num_saison, num_episode \rightarrow adjectif
- 18) mail \rightarrow abonnement
- 19) mail \rightarrow téléphone
- 20) mail \rightarrow mot de passe
- 21) téléphone \rightarrow abonnement
- 22) téléphone \rightarrow mail
- 23) téléphone \rightarrow mot de passe
- 24) téléphone, id_sousCompte \rightarrow profil
- 25) téléphone, id_sousCompte \rightarrow liste
- 26) téléphone, id_sousCompte \rightarrow like
- 27) mail, id_sousCompte \rightarrow profil
- 28) mail, id_sousCompte \rightarrow liste
- 29) mail, id_sousCompte \rightarrow like
- 30) téléphone, id_sousCompte, id_Oeuvre, num_saison, num_episode \rightarrow temps_visionnage
- 31) mail, id_sousCompte, id_Oeuvre, num_saison, num_episode \rightarrow temps_visionnage
- 32) téléphone, id_sousCompte, id_Oeuvre, num_saison, nom_episode \rightarrow temps_visionnage
- 33) mail, id_sousCompte, id_Oeuvre, num_saison, nom_episode \rightarrow temps_visionnage

Après avoir fait la partie 1 du calcul de CV(DF), nous pouvons faire l'étape 2. En calculant toutes les fermetures de tous les attributs de gauche, nous avons observé que l'étape 2 était déjà réalisée, donc aucun changement sur nos dépendances fonctionnelles n'est appliqué.

Enfin, nous avons réalisé l'étape 3. Cela nous donne les dépendances fonctionnelles suivantes :

- (a) $\text{id_Oeuvre} \rightarrow \text{nom_oeuvre}$
- (b) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{nom_episode}$
- (c) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{durée}$
- (d) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{date_sortie}$
- (e) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{distribution}$
- (f) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{categorie_age}$
- (g) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{description}$
- (h) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{genres}$
- (i) $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{adjectif}$
- (j) $\text{mail} \rightarrow \text{téléphone}$
- (k) $\text{téléphone} \rightarrow \text{abonnement}$
- (l) $\text{téléphone} \rightarrow \text{mot de passe}$
- (m) $\text{téléphone}, \text{id_sousCompte} \rightarrow \text{profil},$
- (n) $\text{téléphone}, \text{id_sousCompte} \rightarrow \text{liste}$
- (o) $\text{téléphone}, \text{id_sousCompte} \rightarrow \text{like}$
- (p) $\text{téléphone}, \text{id_sousCompte}, \text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{temps_visionnage}$

2. Regroupement des DF

Enfin, nous pouvons regrouper nos DF pour avoir leur forme finale. Ce qui donne :

- $\text{id_Oeuvre} \rightarrow \text{nom_oeuvre}$
- $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{num_episode}, \text{durée}, \text{date_sortie}, \text{distribution}, \text{categorie_age}, \text{description}, \text{genres}, \text{adjectif}$
- $\text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{durée}, \text{num_episode}, \text{date_sortie}, \text{distribution}, \text{categorie_age}, \text{description}, \text{genres}, \text{adjectif}$
- $\text{mail} \rightarrow \text{abonnement}, \text{téléphone}, \text{mot de passe}$
- $\text{téléphone} \rightarrow \text{abonnement}, \text{mail}, \text{mot de passe}$
- $\text{téléphone}, \text{id_sousCompte} \rightarrow \text{profil}, \text{liste}, \text{like}$
- $\text{mail}, \text{id_sousCompte} \rightarrow \text{profil}, \text{liste}, \text{like}$
- $\text{téléphone}, \text{id_sousCompte}, \text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{temps_visionnage}$
- $\text{mail}, \text{id_sousCompte}, \text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{temps_visionnage}$
- $\text{téléphone}, \text{id_sousCompte}, \text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{temps_visionnage}$
- $\text{mail}, \text{id_sousCompte}, \text{id_Oeuvre}, \text{num_saison}, \text{num_episode} \rightarrow \text{temps_visionnage}$

3. Schéma en 3FN

- (a) R1(id_Oeuvre, nom_oeuvre)
- (b) R2(id_Oeuvre, num_saison, num_episode, nom_episode)
- (c) R3(id_Oeuvre, num_saison, nom_episode, durée, date_sortie, distribution, categorie_age, description, genres, adjectif)
- (d) R5(mail, téléphone)
- (e) R6(téléphone, abonnement, mot_de_passe)
- (f) R7(téléphone, id_sousCompte, profil, liste, like)
- (g) R8(téléphone, id_sousCompte, id_Oeuvre, num_saison, nom_episode, temps_visionnage)
- (h) R9(id_Oeuvre, num_saison, id_sousCompte, num_episode, mail)

2.5.2 Algorithme de décomposition

Rappel de notre Relation R: (id_Oeuvre, nom_Oeuvre, durée, num_saison, num_episode, nom_Episode, date_sortie, genres, distributions, adjectif, description, catégorie_age, profil, like, liste, abonnement, id_sousCompte, mail, téléphone, mot_de_passe, temps_visionnage).

1. Premièrement, nous choisissons la dépendance fonctionnelle non triviale DF1 : $\text{id_Oeuvre} \rightarrow \text{nom_oeuvre}$
2. Ensuite, nous déroulons l'algorithme suivant les méthodes pas à pas.
3. Nous obtenons les relations suivantes :

- (a) Nom_Oeuvre(id_Oeuvre, nom_Oeuvre)
- (b) Oeuvre(id_Oeuvre, num_Saison, num_Episode, nom_Episode, duree, date_Sortie, distribution, categorie_Age, descriptif, genres, adjectif)
- (c) Visionnage(mail, id_sousCompte, id_Oeuvre, num_Saison, num_Episode, temps_visionnage)
- (d) Compte(mail, abonnement, telephone, mot_de_passe)
- (e) Sous_Compte(mail, id_sousCompte, profil, liste, like)
- (f) Cles(mail, id_sousCompten, id_Oeuvre, num_Saison, num_Episode)

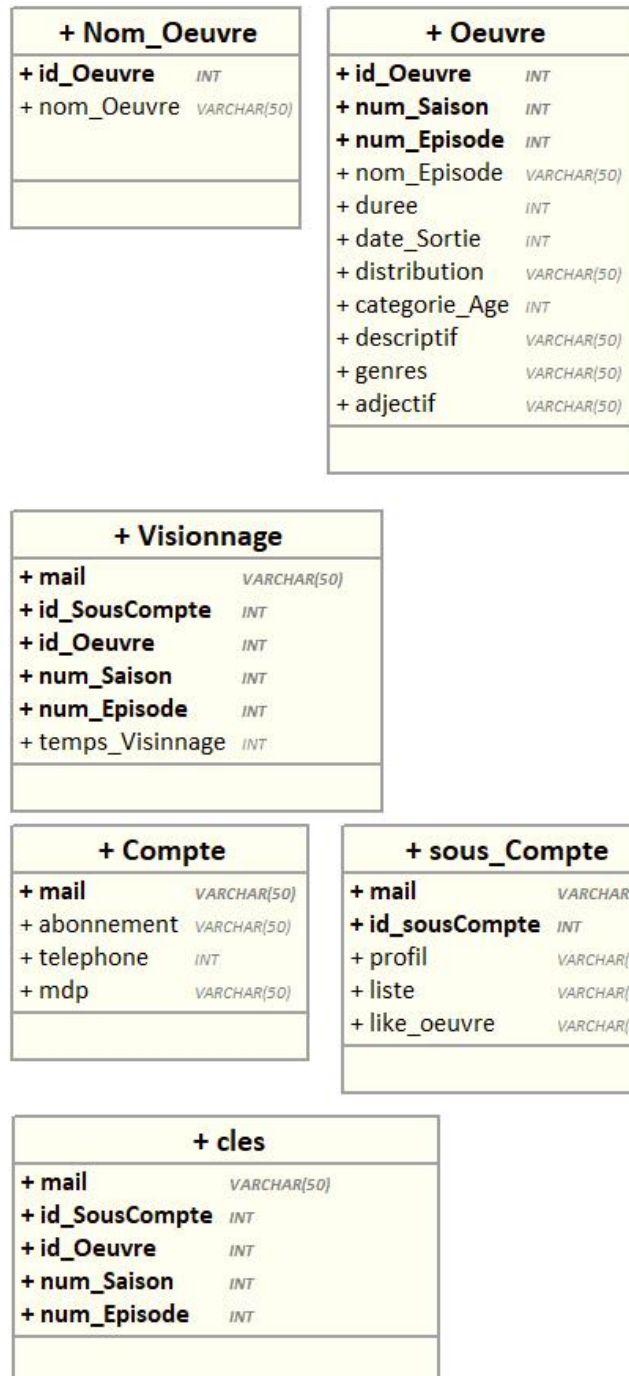
2.5.3 Choix du modèle

Les 2 modèles présentés ci-dessus nous semblent assez similaires. Cependant, nous pensons que le modèle fait avec l'algorithme de décomposition permet de mieux modéliser notre idée de la base de donnée que nous voulons faire et surtout en fonction des requêtes. En effet, nous pensons que la plupart des requêtes faites au niveau des utilisateurs seraient faites avec un mail plutôt qu'avec un numéro de téléphone.

Aussi, nous pensons que les requêtes d'accès au temps de visionnage devraient se faire en fonction du num_Episode, plutôt que de du nom_Episode.

Ainsi, le schéma final présenté cidessous sera celui construit à l'aide de l'algorithme de décomposition.

3 Modèle Relationnel de votre base de données (tables, clés, types, cardinalités, etc).



4 Explication des fonctions, triggers, vues, etc. de votre base de données.

5 Atouts, améliorations possibles et critiques de votre travail.

6 Conclusion.

À l'issue de ce projet, nous avons pu apprendre à faire une base de donnée qui Ce projet nous aura ainsi permis de mettre en application les notions de l'UE étudiées en TD et en CM au sujet des bases de données notamment.

Ce projet aura également été un moyen de parfaire le savoir-faire en L^AT_EX.