

Méthodologie de Conception :

Service proposé : VERSUS

La plateforme permet à l'utilisateur(nom, prénom, sexe, adresse mail, date de naissance, mot de passe, souscription newsletter, liste fournisseurs streaming auquel il est abonné, id utilisateur généré automatique) de pouvoir se connecter à tous les fournisseurs de streaming auxquels il est abonné.

Le but étant de faire du profiling des usagers, leurs goûts en matière de films/séries et d'engranger de la data qui pourrait à posteriori être revendue aux fournisseurs de streaming.

Le service permet à l'utilisateur d'avoir une seule interface sur laquelle il aura accès à tous les films(acteurs, producteurs, réalisateurs) et toutes les séries(acteurs, producteurs, réalisateurs) des plateformes de streaming auxquelles il est abonné et ceci sans changer d'application.

Le profiling permettra d'obtenir des données comme l'historique de visionnage, les heures de visionnage, les plages horaires (voire même les période de l'année) afin de décrire les habitudes de l'utilisateur qui pourra être revendue aux fournisseurs de streaming afin qu'ils puissent proposer du contenu aux usagers.

Le type d'appareils sur lequel l'utilisateur regarde ses contenus sera récupéré afin de personnaliser le contenu proposé.

Observer les données de localisation des utilisateurs afin d'adapter le contenu proposer par le service (films français en France, etc...)

L'âge de l'utilisateur influe sur le contenu proposé.

L'acceptation des CGU permet de récupérer toutes les données que l'utilisateur aura renseigné auprès des fournisseurs de streaming

Un système de notation sera proposé à l'utilisateur, initialement le service récupérera les notations des usagers venant des fournisseurs de streaming, et une base de données interne sera créée lorsque l'utilisateur effectue sa notation directement via VERSUS.

L'utilisateur aura la possibilité d'émettre une critique.

La société gagne de l'argent grâce à la revente de données aux fournisseurs de streaming. De plus, le service proposerait de souscrire à un abonnement auprès des fournisseurs de streaming pour du contenu qui lui conviendrait selon son profiling mais auquel il n'est pas encore abonné.

De plus il y aura une interface de management d'abonnements qui permet à l'utilisateur d'un simple clic de s'abonner ou de se désabonner, avec une sauvegarde des informations de paiement (prix,cb, paypal, adresse de facturation)

Des statistiques mensuelles seront données à l'utilisateur afin de gérer au mieux les abonnements (le fournisseur le plus utilisé, temps de visionnage, genre préféré...)

Un fournisseur favori de l'utilisateur pourra être défini en fonction du nombre de visionnage sur les différentes plateformes afin de rediriger l'utilisateur sur sa plateforme favori dans le cas où un contenu est disponible sur plusieurs plateformes

Un historique de recherche sera stocké afin de pouvoir affiner les recherches ou de proposer un film/série qu'il aurait cherché auparavant qui n'était pas encore disponible avec ses abonnements.

Un tri par catégories sera proposé sur la plateforme pour les films et séries (horreur, comédie, drame...)

Des films peuvent avoir le même titre, ils auront donc un idFilm unique. Quant aux séries,

Type de data :

- Historique de visionnage interplateformes
- Identifier la fourchette d'âge de l'utilisateur
- Les goûts en matière de films/séries
- Heures de visionnage
- Plages horaires, quel jour de la semaine (vacances, périodes de fêtes..)
- Historique de recherche
- Données de localisation
- Les notes données par les utilisateurs
- L'appareil sur lequel le contenu a été regardé
- Watch Time (portion regardée ...)

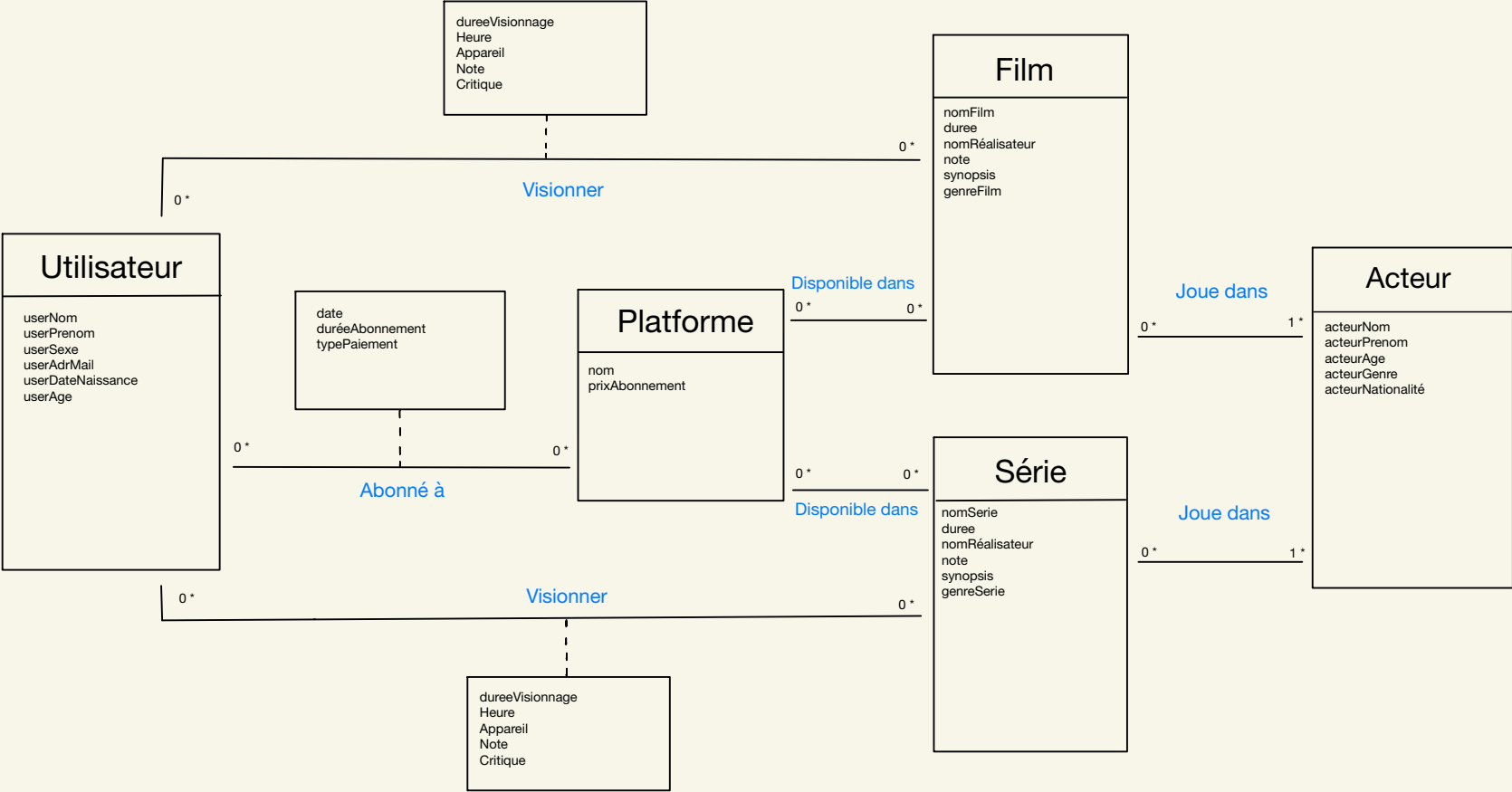
Dictionnaire de données

Libellé	Description	Format	Longueur	Règle de gestion
usager	usager			
cléUsager		AlphaNumérique	25	
nomUsager		Texte	25	
sexeUsager		Texte	1	M ou F
prenomUsager		Texte	25	
mailUsager	adresse mail	AlphaNumérique	25	
dobUsager	date de naissance usager	Date	10	
ageUsager		AlphaNumérique	3	
mdpUsager	mot de passe usager	AlphaNumérique	25	
listUsager	Liste de fournisseurs auxquels l'utilisateur est abonné			Classe
idUsager	numéro unique de l'utilisateur	AlphaNumérique	25	
nbreConnexion	nombre d'utilisateurs connectés à un instant t	Entier		requete
goutUsager	styles de films et séries que l'utilisateur préfère, est-ce qu'il préfère un film ou une série ?			Requête
film	genre, acteurs, durée			Classe
genreFilm		Texte	25	
idFilm				
nomFilm		Texte	25	
nomRéalisateurFilm		Texte	25	

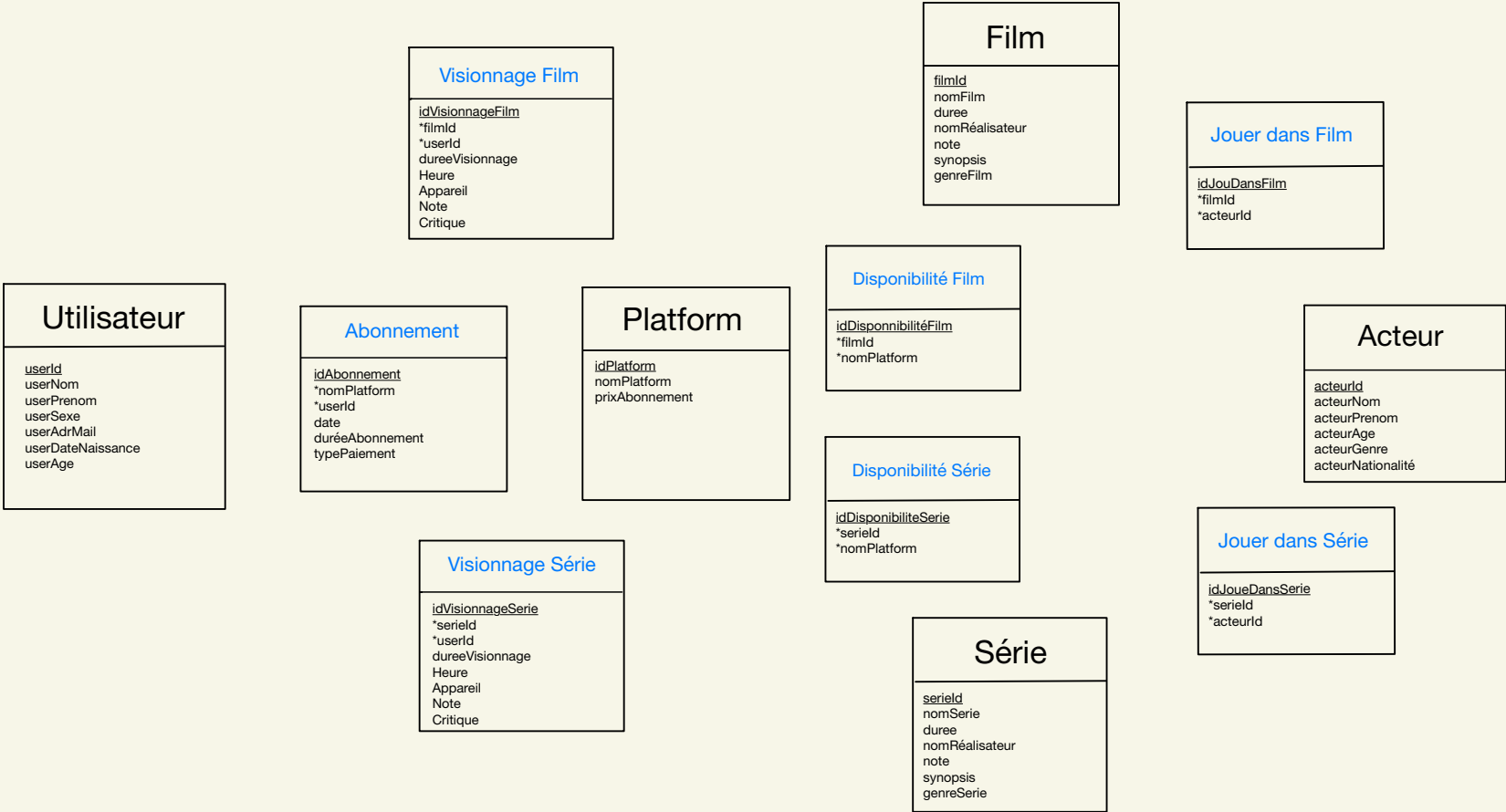
noteFilm		Texte	25	
synopsisFilm		Texte	256	
duréeFilm		Durée		
serie				Classe
genreSérie		Texte	25	
idSérie				
nomSérie		Texte	25	
nomRéalisateurSérie		Texte	25	
noteSérie		Float		
synopsisSérie		Texte	256	
duréeSerie		Durée		
historiqueUsager				Requête
heureH		AlphaNumérique		
dureeVisionnageUsager		AlphaNumérique		
habitudesUsager				Requête
moyensDePaiementUsager				Classe
cbUsager		AlphaNumérique		
paypalUsager		AlphaNumérique		
adresseFacturationUsager		AlphaNumérique		
appareilUtiliseUsager				Classe
telephone		Booléen		
ordinateur		Booléen		
television		Booléen		
tablette		Booléen		
notation		Entier		
notationUtilisateur		Entier		
catalogueFournisseur				
nomFournisseur		AlphaNumérique		
localisationUsager		AlphaNumérique		

nomsActeur		Texte		
nomProducteur		Texte		
nomsRéalisateur		Texte		
synopsisContenu		AlphaNumérique		
critiqueUsager		AlphaNumérique		
PEGI				

Modèle conceptuel de données



Modèle relationnel de données



Normalisation 1-NF

Première étape de la normalisation : nous n'avons pas à modifier nos classes car aucunes d'elles n'as d'attributs non atomiques.

Utilisateur(userId, userNom, userPrenom, userSexe, userAdrMail, userDateNaissance, userAge)

Platform(idPlatform, nomPlatform, prixAbonnement)

Film(filmId, nomFilm, durée, nomRealisateur, note, synopsis, genreFilm)

Acteur(acteurId, acteurNom, acteurPrenom, acteurAge, acteurGenre, acteurNationalite)

Série(serieId, nomSerie, durée, nomRealisateur, note, synopsis, genreSerie)

VisionnageFilm(idVisionnageFilm, filmId*, userId*, dureeVisionnage, Heure, Appareil, Note, Critique)

Abonnement(idAbonnement, nomPlateforme*, userId*, date, duréeAbonnement, typePaiement)

VisionnageSerie(idVisionnageSerie, serieId*, userId*, Durée, Heure, Appareil, Note, Critique)

DisponibilitéFilm(idDisponibilitéFilm, filmId*, nomPlatform*)

DisponibilitéSerie(idDisponibilitéSerie, sérieId*, nomPlatform*)

JouerDansFilm(IdJoueDansFilm, filmId*, nomPlatform*)

JouerDansSerie(IdJoueDansSerie, serieId*, nomPlatform*)

Normalisation 2-NF

Deuxième étape de la normalisation : Nous n'avons pas à modifier nos classes car aucunes d'elles ne possèdent d'attributs non clé qui dépend d'une partie d'une clé.

Utilisateur(userId, userNom, userPrenom, userSexe, userAdrMail, userDateNaissance, userAge)

Platform(idPlatform, nomPlatform, prixAbonnement)

Film(filmId, nomFilm, durée, nomRealisateur, note, synopsis, genreFilm)

Acteur(acteurId, acteurNom, acteurPrenom, acteurAge, acteurGenre, acteurNationalite)

Série(serieId, nomSerie, durée, nomRealisateur, note, synopsis, genreSerie)

VisionnageFilm(idVisionnageFilm, filmId*, userId*, dureeVisionnage, Heure, Appareil, Note, Critique)

Abonnement(idAbonnement, nomPlateforme*, userId*, date, duréeAbonnement, typePaiement)

VisionnageSerie(idVisionnageSerie, serieId*, userId*, Durée, Heure, Appareil, Note, Critique)

DisponibilitéFilm(idDisponibilitéFilm, filmId*, nomPlatform*)

DisponibilitéSerie(idDisponibilitéSerie, sérieId*, nomPlatform*)

JouerDansFilm(IdJoueDansFilm, filmId*, nomPlatform*)

JouerDansSerie(IdJoueDansSerie, serieId*, nomPlatform*)

Normalisation 3-NF

Troisième étape de la normalisation : Nous n'avons pas modifié nos classes car aucunes d'elles ne possèdent d'attribut non clé qui dépend d'un autre attribut non clé.

Utilisateur(userId, userNom, userPrenom, userSexe, userAdrMail, userDateNaissance, userAge)

Platform(idPlatform, nomPlatform, prixAbonnement)

Film(filmId, nomFilm, durée, nomRealisateur, note, synopsis, genreFilm)

Acteur(acteurId, acteurNom, acteurPrenom, acteurAge, acteurGenre, acteurNationalite)

Série(serieId, nomSerie, durée, nomRealisateur, note, synopsis, genreSerie)

VisionnageFilm(idVisionnageFilm, filmId*, userId*, dureeVisionnage, Heure, Appareil, Note, Critique)

Abonnement(idAbonnement, nomPlateforme*, userId*, date, duréeAbonnement, typePaiement)

VisionnageSerie(idVisionnageSerie, serieId*, userId*, Durée, Heure, Appareil, Note, Critique)

DisponibilitéFilm(idDisponibilitéFilm, filmId*, nomPlatform*)

DisponibilitéSerie(idDisponibilitéSerie, sérieId*, nomPlatform*)

JouerDansFilm(IdJoueDansFilm, filmId*, nomPlatform*)

JouerDansSerie(IdJoueDansSerie, serieId*, nomPlatform*)