GRENOBLE EATS

Soutenance orale du projet de Base de Données

Équipe 1 Groupe 2 MMIS :
Hélène HASSAN, Brice PERES, Virgile HENRY, Dimitri HUBANS, Ryan DELAYAT, Zoé DESPREZ, Paul COLINOT

01

Organisation du projet en groupe

Partie THÉORIQUE : Analyse et conception du schéma E/A

2 groupes de travail en parallèle : Paul, Ryan, Dimitri  Hélène, Virgile, Brice

Même travail, réalisé indépendamment des autres

Mise en commun au bout de **~4 séances**

Discussion sur les points difficiles (parties 2 et 3 des slides) → solution commune

Chaque sous-groupe a un regard critique sur l'autre groupe
Tout monde connaît le sujet correctement

Partie PRATIQUE : Conception du code java et des scripts SQL

3 binômes répartis sur 3 aspects du code

- Scripts pour la création (create, clean, fill, reset) de la base de données → Ryan, Dimitri
- Requêtes SQL (lien entre l'application et la base de données) → Paul, Brice
- Interface (application, menus navigables et interactifs) → Virgile, Hélène

2 personnes sur le compte rendu et la relecture de l'analyse → Zoé, Hélène

Tout le monde a une **vision globale** du projet (pratique & théorique) → vie professionnelle
Interactions constantes entre sous-groupes

02

Difficulté n°1 :
Le droit à l'oubli

Comment implémenter le droit à l'oubli ?

Cahier des charges:

- Pouvoir supprimer toute les données confidentielles d'un client
- Conserver les informations sur les commandes passées

Comment identifier un client ? Un client supprimé ?

IdClient :

- identifier un client sans données personnelles

InfoClients :

- Rattaché à un ID Client
- Contient les informations du client
- Peut être supprimé

Comment implémenter le droit à l'oubli ?

Différentes solutions envisageables:

- Avoir 2 types de clients (supprimés et non supprimés)

A l'aide de sous entités (OU, obligatoire)

Non: car ceux supprimés n'ont aucun attributs associés

- Stocker un idClient auquel serait potentiellement rattachés les informations du clients si elle n'ont pas été oubliées

A l'aide d'une entité faible

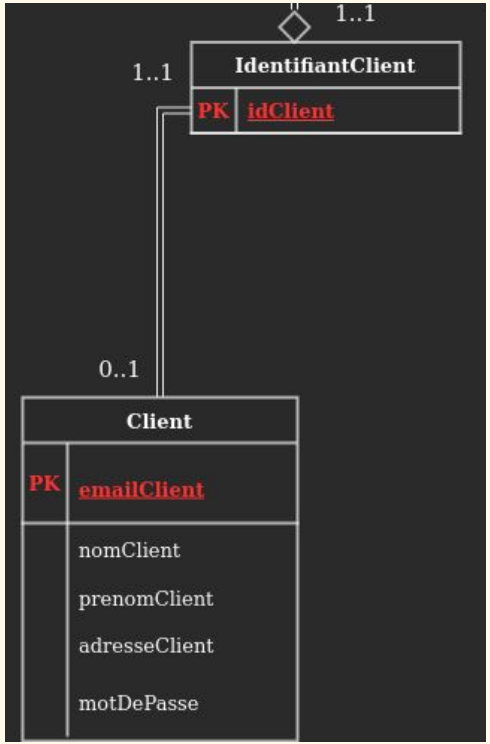
Non: Car les informations d'un client se suffisent à elle même pour l'identifier

Comment implémenter le droit à l'oubli ?

Solution adoptée:

- 2 entités informationClient et identifiantClient

Toutes les informations des commandes se réfèrent alors à identifiantClient
-> Client peut être supprimé
-> Droit à l'oubli respecté



03

Difficulté n°2 : Les sous-types de commandes

Étape 1 : analyse statique

Une commande peut être : **à emporter**, **sur place**, **en livraison**

Identifiée de **manière unique**
(identifiant de commandes → *idCommande, idLivraison, idSurPlace*)

→ {idLivraison} ∪ {idSurPlace} ⊆ {idCommande}

Ext(idLivraison) ⊆ Ext(idCommande)
Ext(idSurPlace) ⊆ Ext(idCommande)

Une commande ne peut être à la fois en livraison et sur place : → {idLivraison} ∩ {idSurPlace} = ∅

Contient obligatoirement : *dateCommande, heureCommande, idClient, emailRestaurant, typeCommandeClient, contenu, prixFinal, statut*

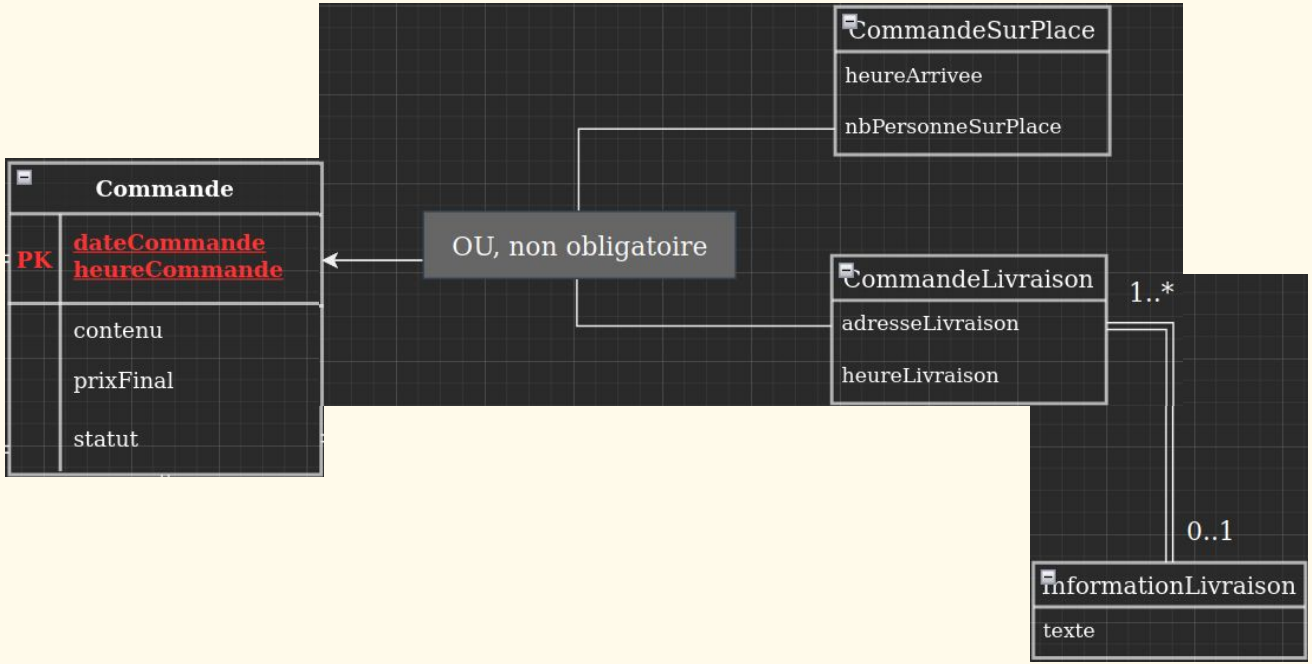
→ ensemble d'attributs hérités d'*idCommande* dans une **DF** + contraintes de valeurs non spécifiées ici

Une commande en livraison (*idLivraison*) doit **EN PLUS contenir obligatoirement** : *adresseLivraison* et **optionnellement :** *texteLivraison*

→ *idLivraison* → *adresseLivraison* (***DF***)

→ *idLivraison* -|→ *texteLivreur* et *texteLivreur* ->> *idLivraison* (**Contrainte de multiplicité)**

Étape 2 : passage au schéma Entités/Associations



Étape 3 : passage au relationnel - quels choix possibles ?

~~Choix 1 : Duplication~~ → CONSOMME TROP

Choix 2 : Référence

PAS FN1 ← ~~Choix 3 : Unification~~

Commandes(emailRestaurant, idClient, dateCommande, heureCommande, prixFinal, statut)

CommandesLivraison(emailRestaurant, idClient, dateCommande, heureCommande, prixFinal, statut, adresseLivraison, heureLivraison)

CommandesSurPlace(emailRestaurant, idClient, dateCommande, heureCommande, prixFinal, statut, nbPersonnesSurPlace, heureArrivee)

+ Accès simple et immédiat aux n-uplets des sous-commandes
+ Facilité d'expression des contraintes
- Double insertion et empreinte mémoire++
- Mises à jour parfois pénibles

Commandes(emailRestaurant, idClient, dateCommande, heureCommande, prixFinal, statut)

CommandesLivraison(emailRestaurant, idClient, dateCommande, heureCommande, adresseLivraison, heureLivraison)

CommandesSurPlace(emailRestaurant, idClient, dateCommande, heureCommande, nbPersonnesSurPlace, heureArrivee)

+ Facilité d'expression des contraintes
+ Moins d'empreinte mémoire
+ Pas de problème de mises à jour
- Double insertion
- Accès aux n-uplets des sous-commandes + coûteux (jointures)

Commandes(emailRestaurant, idClient, dateCommande, heureCommande, prixFinal, statut, adresseLivraison, heureLivraison, nbPersonnesSurPlace, heureArrivee, TypeCommandesClient)

⚠ Mettre les bons attributs='null' selon la valeur de TypeCommandesClient

+ Insertion unique
+ Accès aux n-uplets immédiat
+ Pas de problème de mises à jour
- Expression de contraintes de référence vers les sous-commandes gérée par l'application

Autres éléments difficiles (non détaillés ici)

- Propriété propre ‘Quantité’ pour les **contenus de commande**
- Arbre des catégories de cuisine

Détaillés dans le rapport !

04

L'application

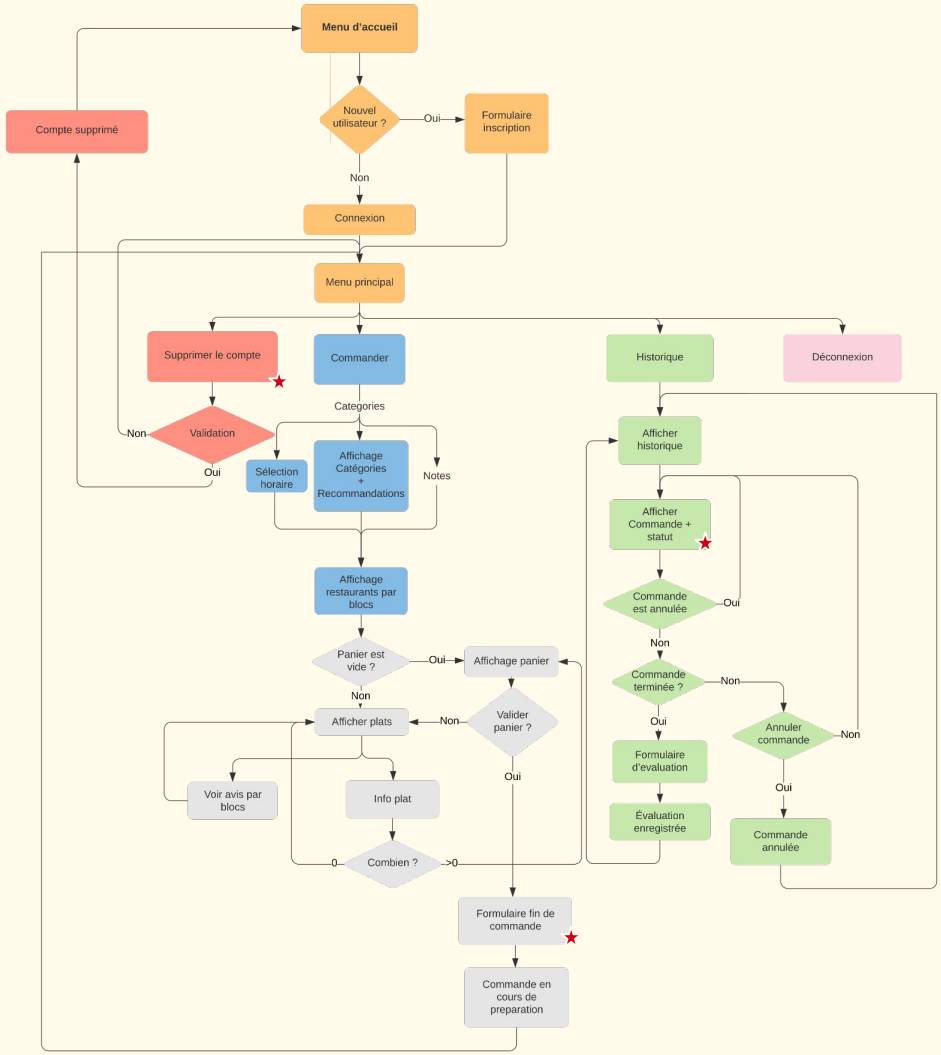
Automate pour la prise de décisions

Définit un **cahier des charges** pour notre application

un gage de qualité pour le client !

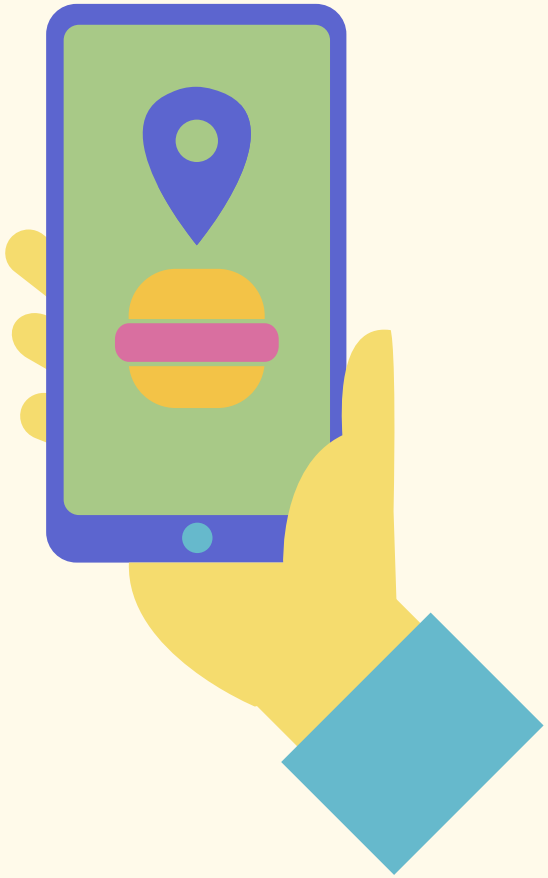
Partie technique (transactions et requêtes SQL) présentes dans le compte rendu !

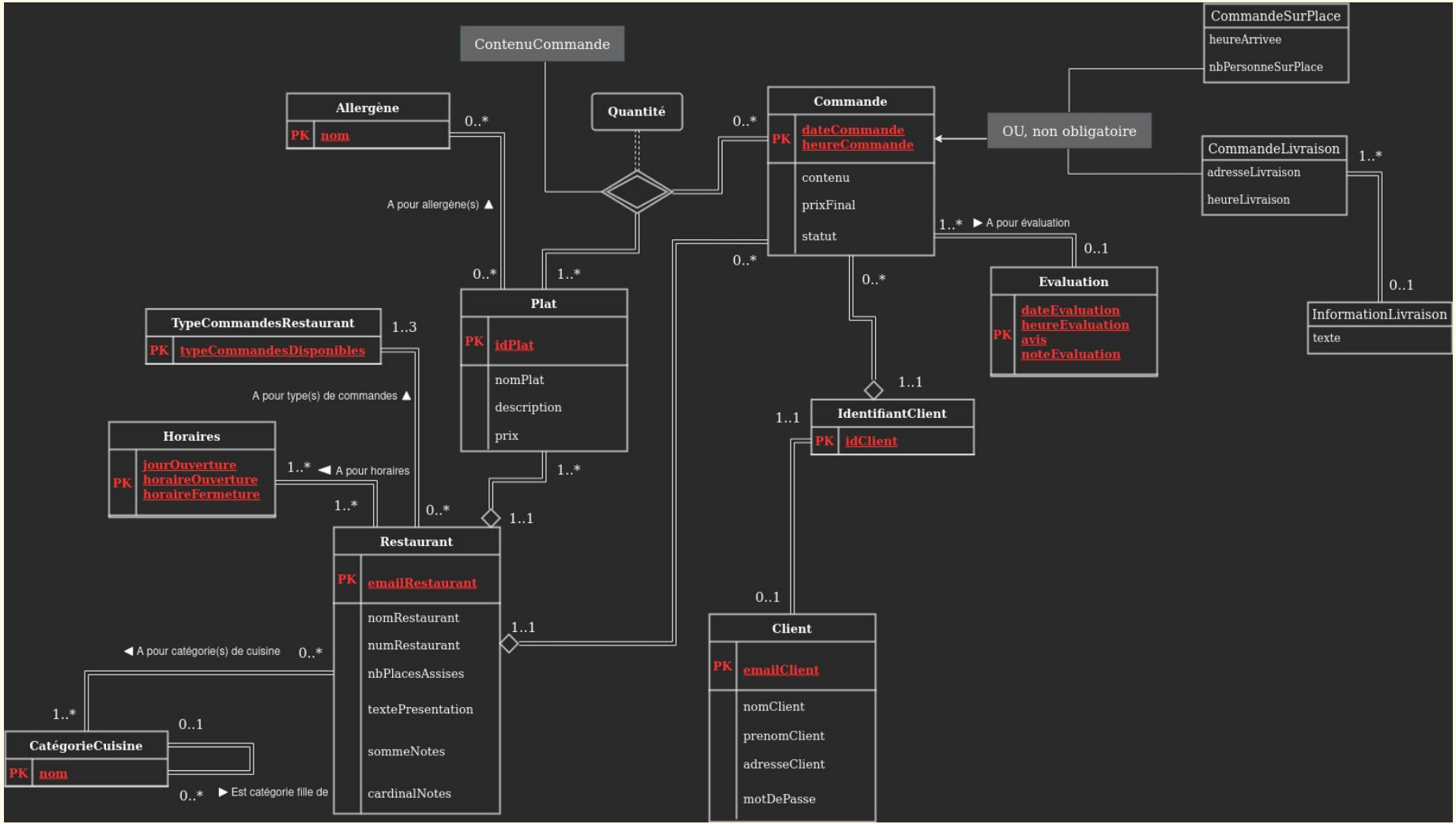
★ transactions



Conclusion

Merci pour votre écoute !




ContenuCommande
CommandeSurPlace
heureArrivee
nbPersonneSurPlace
Allergène
PK nom
0..*
Quantité
0..*
Commande
PK dateCommande
heureCommande
contenu
prixFinal
statut
OU, non obligatoire
CommandeLivraison
adresseLivraison
heureLivraison
1..*
A pour allergène(s) ▲
0..*
1..*
1..* ▶ A pour évaluation
0..1
0..*
0..*
Plat
PK idPlat
nomPlat
description
prix
Evaluation
PK dateEvaluation
heureEvaluation
avis
noteEvaluation
0..1
InformationLivraison
texte
TypeCommandesRestaurant
PK typeCommandesDisponibles
1..3
A pour type(s) de commandes ▲
1..1
IdentifiantClient
PK idClient
1..1
Horaires
PK jourOuverture
horaireOuverture
horaireFermeture
1..* ◀ A pour horaires
1..*
1..*
0..*
1..1
Restaurant
PK emailRestaurant
nomRestaurant
numRestaurant
nbPlacesAssises
textePresentation
sommeNotes
cardinalNotes
1..1
0..1
Client
PK emailClient
nomClient
prenomClient
adresseClient
motDePasse
◀ A pour catégorie(s) de cuisine
0..*
1..*
0..1
CatégorieCuisine
PK nom
0..*
▶ Est catégorie fille de