

Projet Base de données

DOSSIER PROJET

Projet base de donnée

Contexte :

Giselle souhaite vendre ses potions, les ingrédients et des onguents via un site internet. Au niveau des ingrédients, ils seront tous fournis par des fournisseurs, ces ingrédients doivent rester frais. Le type des ingrédients et le type de magie ne préoccupent plus Giselle. Au niveau des commandes, elle souhaite que les clients choisissent le récipient de leur potion.

Besoin :

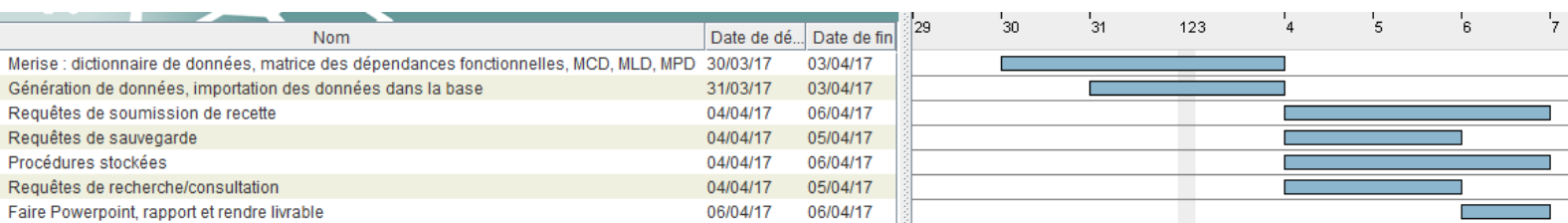
Giselle nous a contacté afin de créer la base de données de ses potions ainsi que des requêtes de gestion des données de la base, de soumission de nouvelle recette, des requêtes de recherche/consultation et de sauvegarde.

Présentation du groupe :

-Gildas COUSIN (Chef de projet)

-Hugo DELTOUR

Planning :



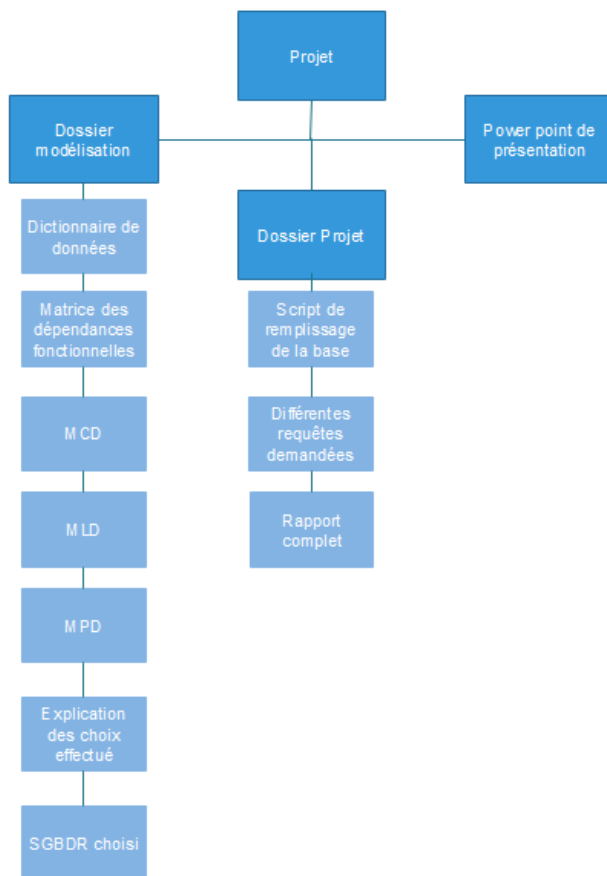
Outils utilisés :

Afin de réaliser ce projet, nous avons eu recours à MySQL workbench, MySQLDump, phpMyAdmin, GanttProject, E-Draw Max, Trello et JMerise. Nous avons aussi créé un drive sur Google Drive afin de faciliter l'échange de fichier au sein du groupe.

Projet base de donnée

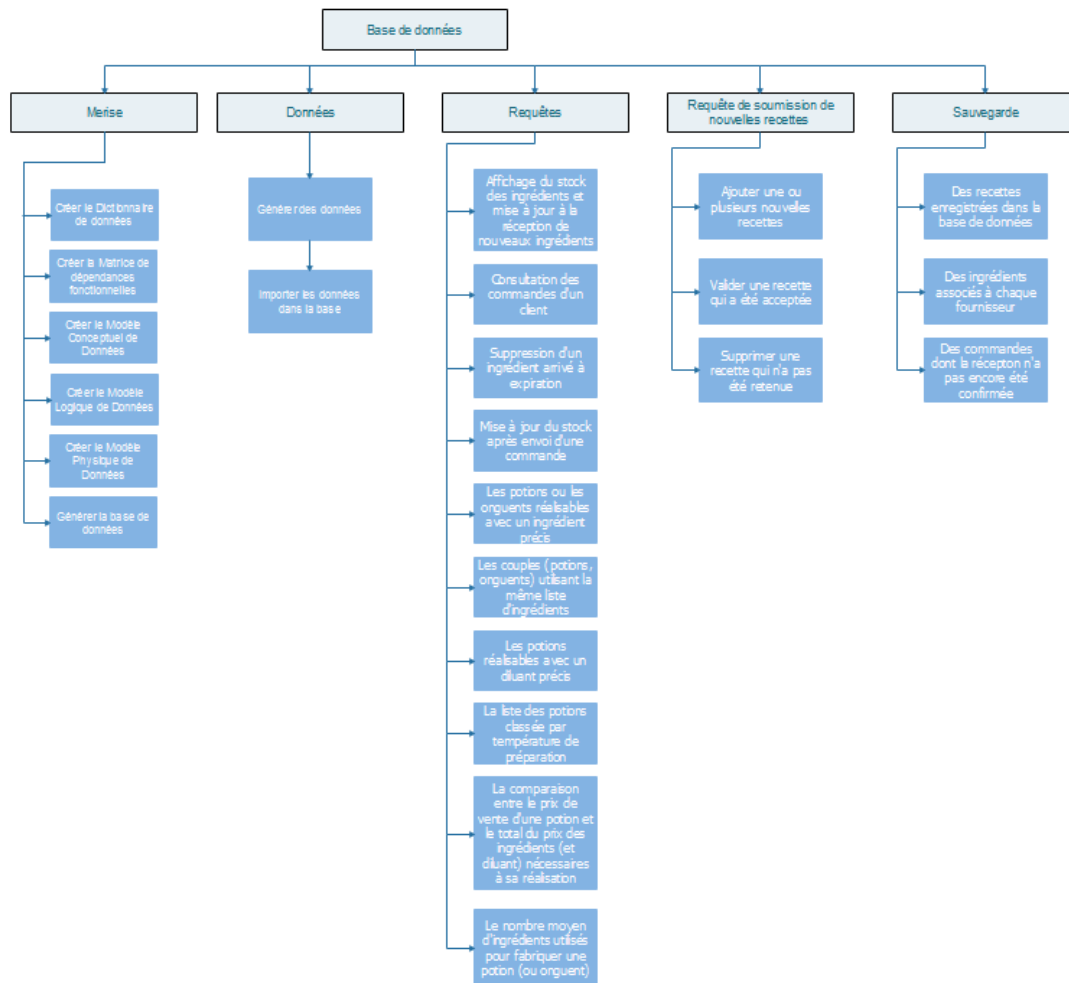
Gestion de Projet :

Notre PBS (Product Breakdown Structure):



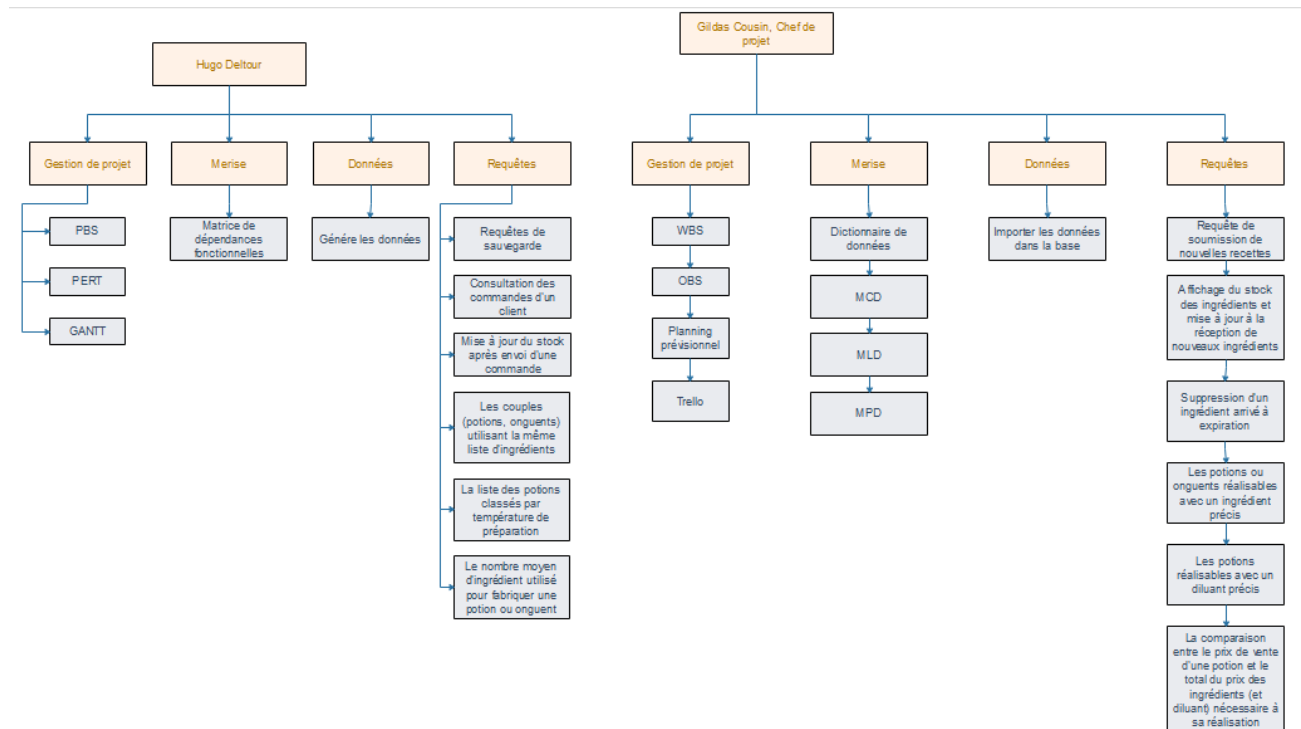
Projet base de donnée

Notre WBS (Work Breakdown Structure):

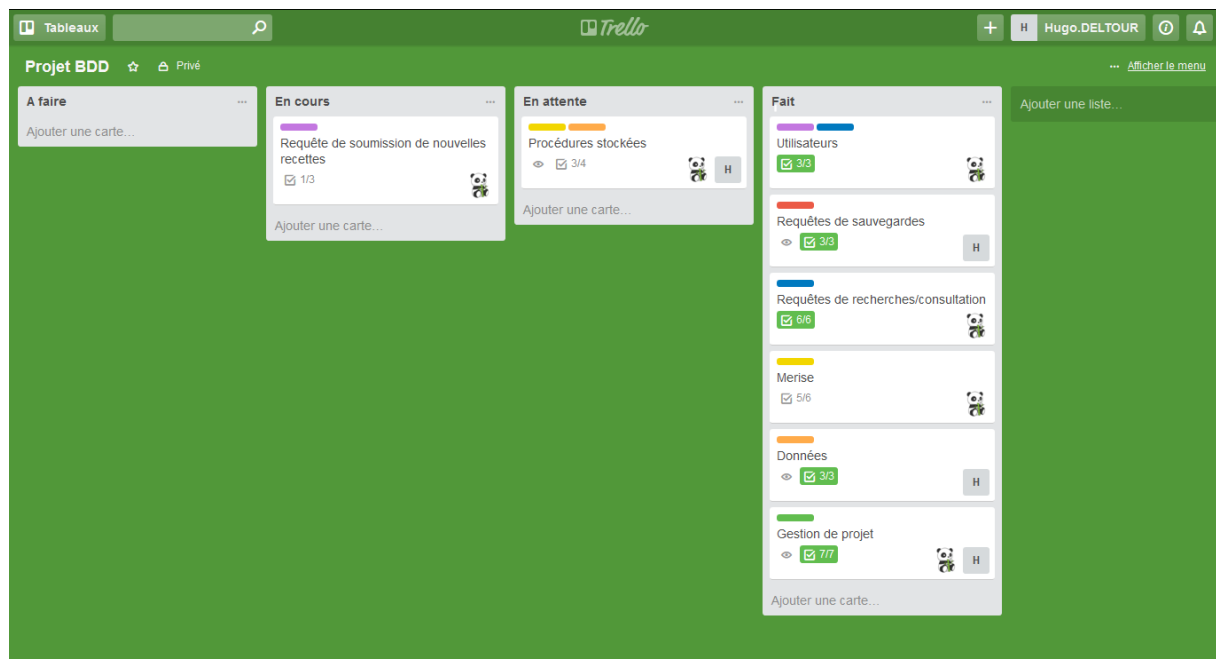


Projet base de donnée

Notre OBS (Organizational Breakdown Structure):



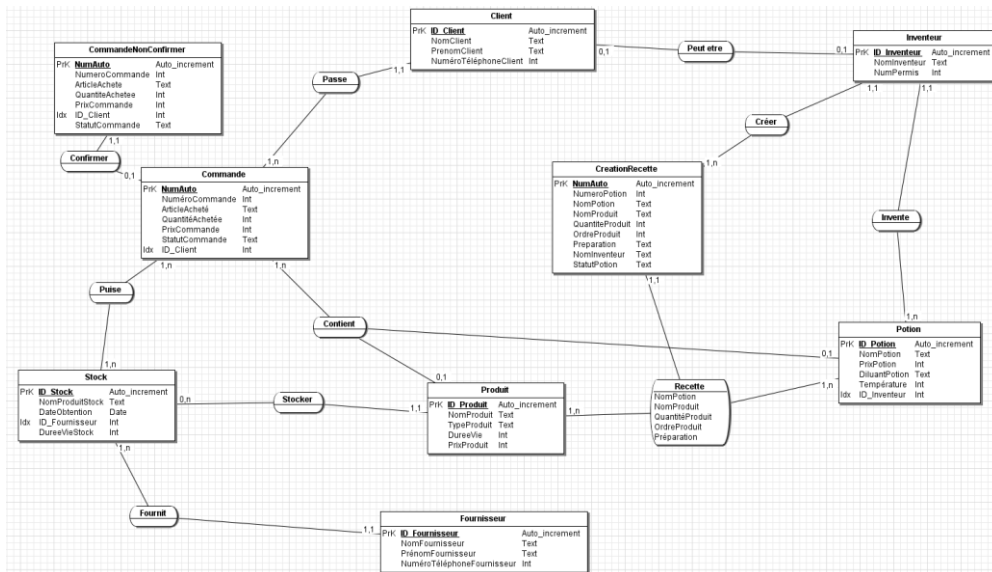
Notre Trello :



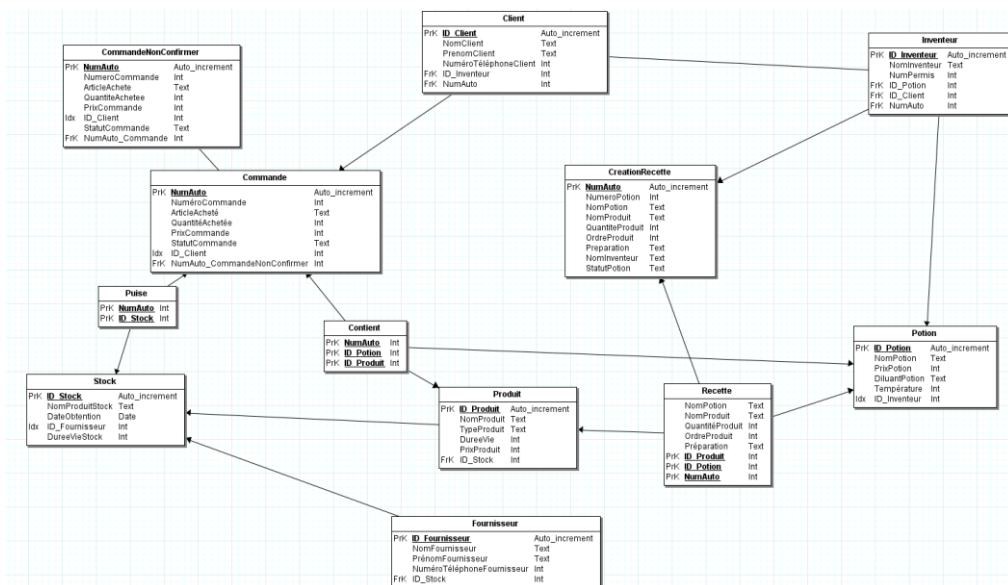
Projet base de donnée

Ecart avec le dossier modélisation :

MCD (Model Conceptuel de donnée) modifié :



MLD (Model Logique de Donnée) modifié :



Projet base de donnée

Nous avons modifié le MCD et le MLD puisque nous avons remarqué qu'il manquait des tables et champs lors de la création de nos requêtes.

Génération des données :

Pour générer les données de la table, nous avons pris les ingrédients du jeu Skyrim, les nom, prénom et numéro de téléphone des client et fournisseur viennent du site fr.fakenamegenerator.com et nous avons-nous même créer les recettes des potions ainsi que leur nom.

Requêtes à faire :

Nous avons à faire différentes requêtes afin de répondre aux attentes de Giselle. Nous devons faire les requêtes de recherche et consultation de la base qui sont découpée en 6 requêtes :

- Les potions ou les onguents réalisable avec un ingrédient précis ;
- Les couples (potions, onguents) utilisant la même liste d'ingrédients ;
- Les potions réalisables avec un diluant précis ;
- La liste des potions classée par température de préparation ;
- La comparaison entre le prix de vente d'une potion et le total du prix des ingrédients (et diluant) nécessaire à sa réalisation ;
- Le nombre moyen d'ingrédients utilisés pour fabriquer une potion (ou onguent) ;

Nous devons aussi faire des requêtes de soumission de nouvelle recette :

- Ajouter une ou plusieurs nouvelles recettes ;
- Valider une recette qui a été acceptée ;
- Supprimer une recette qui n'a pas été retenue ;

Nous devons également faire des procédures stockées pour les requêtes de gestion de base de données :

- Affichage du stock des ingrédients et mise à jour à la réception de nouveaux ingrédients ;
- Consultation des commandes d'un client ;
- Suppression d'un ingrédient arrivé à expiration ;
- Mise à jour du stock après envoi d'une commande ;

Projet base de donnée

Nous avons à sauvegarder certaines données de la base :

- Recettes enregistrées dans la base de données ;
- Ingrédient associés à chaque fournisseur ;
- Commandes dont la réception n'a pas encore été confirmée ;

Réalisation :

Pour la réalisation, nous avons fait des requêtes sous formes de procédures stockées :

- Les potions ou les onguents réalisable avec un ingrédient précis :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `IngrédientPrécis` (IN `NomIngrédientChoisi` TEXT)
BEGIN
```

```
SELECT recette.ID_Potion, recette.NomPotion, recette.NomProduit
```

```
FROM recette
```

```
WHERE recette.NomProduit = NomIngrédientChoisi;
```

```
END |
```

DELIMITER ;

- Les couples (potions, onguents) utilisant la même liste d'ingrédients :

- Les potions réalisables avec un diluant précis :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `IngrédientPrécis` (IN `NomIngrédientChoisi` TEXT)
BEGIN
```

```
SELECT recette.ID_Potion, recette.NomPotion, recette.NomProduit
```

```
FROM recette
```

```
WHERE recette.NomProduit = NomIngrédientChoisi;
```

```
END |
```

DELIMITER ;

Projet base de donnée

-La liste des potions classée par température de préparation :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `PotionTemperature`()
```

```
BEGIN
```

```
SELECT potion.NomPotion, potion.Temperature
```

```
FROM potion
```

```
WHERE potion.NomPotion LIKE 'Potion%'
```

```
ORDER BY potion.Temperature ASC;
```

```
END |
```

DELIMITER ;

-La comparaison entre le prix de vente d'une potion et le total du prix des ingrédients (et diluant) nécessaire à sa réalisation :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `ComparaisonPrixPotion`(IN `PotionChoisie` TEXT)
BEGIN
```

```
SELECT SUM(produit.PrixProduit) AS PrixTotalIngrédient
```

```
FROM produit
```

```
NATURAL JOIN recette
```

```
WHERE recette.NomPotion = PotionChoisie;
```

```
SELECT DISTINCT potion.PrixPotion AS PrixTotalPotion
```

```
FROM potion
```

```
NATURAL JOIN recette
```

```
WHERE recette.NomPotion = PotionChoisie;
```

```
SELECT PrixTotalPotion - PrixTotalIngrédient;
```

```
END |
```

Projet base de donnée

-Le nombre moyen d'ingrédients utilisés pour fabriquer une potion (ou onguent) :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `NombreMoyenIngredient` ()
```

```
BEGIN
```

```
SELECT COUNT (*) / 25
```

```
FROM recette;
```

```
END |
```

DELIMITER;

-Ajouter une ou plusieurs nouvelles recettes :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `AjoutRecette` (IN NumeroRecette int, IN  
NomNouvellePotion text, IN NomProduitRecette text, IN QuantiteProduitRecette int, IN  
OrdreProduitRecette int, IN PreparationRecette text, IN NomInventeurPotion text)
```

```
BEGIN
```

```
INSERT INTO creationrecette
```

```
(NumeroPotion,NomPotion,NomProduit,QuantiteProduit,OrdreProduit,Preparation,NomInventeur)
```

```
VALUES
```

```
(NumeroRecette,NomNouvellePotion,NomProduitRecette,QuantiteProduitRecette,OrdreProduitRece  
tte,PreparationRecette,NomInventeurPotion);
```

```
END |
```

-Valider une recette qui a été acceptée :

-Supprimer une recette qui n'a pas été retenue :

Projet base de donnée

-Affichage du stock des ingrédients et mise à jour à la réception de nouveaux ingrédients :

```
DELIMITER $$ CREATE DEFINER=`root`@`localhost` PROCEDURE `AffichageStock`()  
  
BEGIN  
  
SELECT*  
  
FROM stock;  
  
END |  
  
DELIMITER;
```

-Consultation des commandes d'un client :

```
DELIMITER |  
  
CREATE DEFINER=`root`@`localhost` PROCEDURE `ConsultationCommande`(IN  
ChoixNumeroCommande INT)  
  
BEGIN  
  
SELECT commande. *  
  
FROM commande  
  
WHERE commande.NumeroCommande = ChoixNumeroCommande;  
  
END |  
  
DELIMITER ;
```

-Suppression d'un ingrédient arrivé à expiration :

```
CREATE VIEW FraicheurStock AS  
  
SELECT stock.NomProduitStock, DATEDIFF(NOW(), stock.DateObtention) as JourEnStock,  
stock.DureeVieStock, IF (DATEDIFF(NOW(), stock.DateObtention)>DureeVieStock, 'Pourri', 'Frais') as  
Fraicheur  
  
FROM stock;  
  
DELIMITER |  
  
CREATE DEFINER=`root`@`localhost` PROCEDURE `SuppressionIngrédientPourri`()  
  
BEGIN  
  
DELETE FROM stock  
  
WHERE DATEDIFF(NOW (), stock.DateObtention)>DureeVieStock;
```

Projet base de donnée

END|

DELIMITER;

-Mise à jour du stock après envoi d'une commande :

-Sauvegarde des recettes enregistrées dans la base de données, des ingrédients associés à chaque fournisseur et des commandes dont la réception n'a pas encore été confirmée :

Pour c'est requêtes il nous a été demandé de les faire avec mySQLdump. De ce fait, nous devons passer par l'inviter de commande de windows mais avant cela nous devons trier les commandes en fonction de leur statut et écarter celle qui sont confirmées :

Tout d'abord, il nous faut créer une nouvelle table afin de mettre les commandes résultantes de la prochaine requêtes :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `CommandeNonConfirmer`()
```

```
BEGIN
```

```
INSERT INTO CommandeNonConfirmer
```

```
(NumAuto,NumeroCommande,ArticleAchete,QuantiteAchete,PrixCommande,ID_Client,StatutCom  
mande)
```

```
SELECT *
```

```
FROM commande
```

```
WHERE StatutCommande!="Confirmer";
```

```
END|
```

DELIMITER ;

Ensuite, effectuons la requête de tri des commandes :

DELIMITER |

```
CREATE DEFINER='root'@'localhost' PROCEDURE `CommandeNonConfirmer`()
```

```
BEGIN
```

```
INSERT INTO CommandeNonConfirmer
```

```
(NumAuto,NumeroCommande,ArticleAchete,QuantiteAchete,PrixCommande,ID_Client,StatutCom  
mande)
```

```
SELECT *
```

Projet base de donnée

FROM commande

WHERE StatutCommande!="Confirmer";

END|

DELIMITER ;

Passons maintenant sur l'invité de commande :

En premier lieu, il faut revenir à la racine C:\ .

Ensuite, il faut trouver le chemin d'accès vers mysqldump.exe en règle générale le chemin sera : C:\Program Files\MySQL\MySQL Server 5.7\bin

Maintenant revenons sur notre invité de commande et entrons la commande : cd « C:\Program Files\MySQL\MySQL Server 5.7\bin »

A présent, nous pouvons effectuer notre sauvegarde vers le fichier voulu grâce à la commande :

```
mysqldump.exe -u 'NomUtilisateur' -p --single-transaction --databases 'NomBaseDonnée' --  
tables 'TableàSauvegarder' > "CheminDuDossierDeSauvegarde\FichierDeSauvegarde.sql"
```

En remplaçant les mots entre guillemet par le nom d'utilisateur de la base de données, les chemins, base de données ou tables ,devant être sauvegarder, voulu .

Nous pouvons dorénavant réimporter nos tables sauvegardées vers une base de données vide ou non grâce à la commande :

```
mysql -u 'NomUtilisateur' -p 'BaseDonnéeVide(ou non)' <  
« CheminDuDossierDeSauvegarde\FichierDeSauvegarde.sql"
```

En remplaçant les mots entre guillemet par le nom d'utilisateur de la base de données, les chemins, base de données ou tables ,devant être sauvegarder, voulu .

Bilan projet :

Dans l'ensemble, le projet s'est bien déroulé malgré un retard lors de la création des données. Cependant nous avons quelques écarts concernant les requêtes qui n'ont pu être réalisées.

Bilan groupe :

Malgré quelque friction au sein du groupe, nous avons réussi à avoir un point d'entente et ainsi réaliser le projet. Nous avons eu quelque moment de dilemme au cours du projet ce qui a entraîné quelque écarts mineurs sur les requêtes.