

**Rapport de projet**

**Modélisation**

**& BDD**

# **Table des matières**

[I. Table des matières 2](#_Toc511117723)

[II. Table des figures 3](#_Toc511117724)

[III. Contexte 4](#_Toc511117725)

[IV. Analyse du besoin 4](#_Toc511117726)

[V. Gestion de projet 4](#_Toc511117727)

[Equipe 4](#_Toc511117728)

[Diagramme WBS 5](#_Toc511117729)

[Matrice RACI 5](#_Toc511117730)

[Trello 6](#_Toc511117731)

[Diagramme de GANTT 7](#_Toc511117732)

[VI. Logiciels & services utilisés 8](#_Toc511117733)

[VII. Merise 9](#_Toc511117734)

[Dictionnaire de données 9](#_Toc511117735)

[Matrice des Dépendances Fonctionnelles 10](#_Toc511117736)

[Modèle Conceptuel de Données 11](#_Toc511117737)

[VIII. Implémentation de la BDD 15](#_Toc511117738)

[IX. Requêtes 15](#_Toc511117739)

[Requête I : « Divinité préférée par province » 15](#_Toc511117740)

[Requête II : « Prix moyen des équipement » 15](#_Toc511117741)

[Requête III : « Paiement en Oo ou Oa par province » 16](#_Toc511117742)

[Requête IV : « Comparaison ventes provinces & Egypte » 16](#_Toc511117743)

[Requête V : «  Comparaison ventes avec et sans fête religieuse » 16](#_Toc511117744)

[Requête VI : « Comparaison ventes avec et sans guerre » 17](#_Toc511117745)

[Requête VII : « Prix de vente artisan selon la divinité » 17](#_Toc511117746)

[X. Procédure 17](#_Toc511117747)

[XI. Vue 18](#_Toc511117748)

[XII. Boîte à moustache 18](#_Toc511117749)

[XIII. Bilan personnel 19](#_Toc511117750)

[Valérian POIRIER 19](#_Toc511117751)

[Simon MANIEZ 20](#_Toc511117752)

[Paul FICOT 20](#_Toc511117753)

# Table des figures

[**Figure 1** : Présentation de l'équipe 4](#_Toc511081721)

[**Figure 2** : Work Breakdown Structure 5](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081722)

[**Figure 3** : Matrice RACI 5](#_Toc511081723)

[**Figure 4** : Trello 6](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081724)

[**Figure 5** : Gantt réel 7](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081725)

[**Figure 6** : Gantt prévisionnel 7](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081726)

[**Figure 7** : Dictionnaire de données 9](#_Toc511081727)

[**Figure 8** : Exemple table individu 10](#_Toc511081728)

[**Figure 9** : Matrice des Dépendances Fonctionnelles [partie 1] 10](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081729)

[**Figure 10** : Correspondance couleurs MDF 10](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081730)

[**Figure** **11** : Matrice des Dépendances Fonctionnelles [partie 2] 11](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081731)

[**Figure 12** : MCD Total 11](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081732)

[**Figure 13** : Haut gauche MCD 12](#_Toc511081733)

[**Figure 14** : Haut droite MCD 13](#_Toc511081734)

[**Figure 15** : MCD Bas 14](#_Toc511081735)

[**Figure 16** : Requête I 15](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081736)

[**Figure 17** : Requête II 15](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081737)

[**Figure 18** : Requête III 16](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081738)

[**Figure 19** : Requête IV 16](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081739)

[**Figure 20** : Requête V 16](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081740)

[**Figure 21** : Requête VI 17](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081741)

[**Figure 22** : Procédure stockée 17](#_Toc511081742)

[**Figure 23** : Vue simplifiée 18](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081743)

[**Figure 24** : Boîte à moustache 1 18](file:///P:\CESI.exia\PROJETS\Modélisation%20et%20BDD\Rapport%20MANIEZ%20POIRIER%20FICOT.docx#_Toc511081744)

[**Figure 25** : Boîte à moustache 2 18](#_Toc511081745)

[**Figure 26** : Boîte à moustache 3 19](#_Toc511081746)

[**Figure 27** : Boîte à moustache 4 19](#_Toc511081747)

# Contexte

Héphaïstos le dieu du feu et de la forge souhaite agrandir sa production d’armes à l’Egypte pour ce faire, Hermès prend en main la gestion des ventes et souhaite réaliser un Trapèzis pour gérer ces ventes.

# Analyse du besoin

* Implémenter les données des registres dans une base de données
* Réaliser un outil permettant d’enregistrer une vente automatiquement
* Réaliser des requêtes SQL afin d’obtenir diverses informations sur la base de données

# Gestion de projet

## Equipe

**Figure 1** : Présentation de l'équipe

L’équipe fut composée de Valérian POIRIER, il a travaillé sur les différentes **requêtes SQL**, la **procédure stockée**, la **vue** ainsi que la **boîte à moustache**.

De Simon MANIEZ, ce dernier a travaillé sur l’implantation des données sous forme de registre **Microsoft Excel** vers une base de données **MySQL**.

J’ai occupé pendant ce projet le rôle de chef de projet et me suis chargé de l’étude **Merise**, ainsi que de l’organisation du projet grâce à **Trello**, **Gantt**, matrice **RACI** et **WBS**.

## Diagramme WBS

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevéPour la gestion de ce projet nous avons tout d’abord défini le périmètre du projet grâce à un Work Breakdown Structure(**WBS**).

**Figure 2** : Work Breakdown Structure

Cet organigramme nous a permis de définir les différentes tâches à réaliser dans ce projet.

## Matrice RACI

Puis nous avons utilisés la matrice RACI qui permet de représenter les responsabilités de chacun dans les différentes tâches à réaliser dans ce projet.

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

**Figure 3** : Matrice RACI

## Trello

Nous avons également utilisé l’outils en ligne Trello qui permet de répertorier les tâches en trois colonnes :

* À faire
* En cours
* Fait

Une image contenant extérieur, texte

Description générée avec un niveau de confiance élevéCet outil est utile pour classifier les tâches et être renseigné en temps réel de l’avancement du projet.

**Figure 4** : Trello

## Diagramme de GANTT

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevéUne image contenant capture d’écran

Description générée avec un niveau de confiance très élevéLe diagramme de GANTT est un **outil d’ordonnancement et de gestion de projet**, cela permet de donner une aperçue rapide du déroulement du projet et des tâches réalisés par chacun des membres du projet. Le seul bémol de ce diagramme est la précision qu’il apporte, en effet ce diagramme est précis « haut jour près »

**Figure 5** : Gantt réel

**Figure 6** : Gantt prévisionnel

On peut voir qu’il y a une grosse différence entre le Gantt prévisionnel et le réel, Simon s’est occupé de l’implémentation pendant une grande partie du projet, et ce à cause de problèmes de clés étrangères et de doublons inexistants dans la table.

Et j’ai un peu aidé Valérian avec un script ainsi que la vue.

# Logiciels & services utilisés

Pour réaliser notre solution nous avons utilisé différents outils et logiciels :

Trello est un outil de **gestion de projet** en ligne qui nous a permis de gérer la réalisation globale du projet et des taches de chaque personne du groupe.

Microsoft Access est un **outil de la suite Microsoft** permettant de réaliser des bases de données.

Microsoft Excel est un **outil de la suite Microsoft** permettant de réaliser des tableurs, ainsi que des bases de données (assez restreint).

Microsoft Teams est un **outil de la suite Microsoft** permettant de créer un groupe de discussion instantanée et de partage de données avec une équipe sélectionnée auparavant dans le logiciel.

Tom’s planner est un outil en ligne ayant des **fonctionnalités similaires à Gantt Project**.

En effet cet outil a permis de réaliser le diagramme de Gantt du projet.

MySQL Workbench est un logiciel permettant de réaliser des bases de données ainsi que des requêtes, procédures, et vues en SQL dans un serveur local héberger sur le PC.

JMerise est un logiciel permettant de réaliser une étude de BDD avec la méthode **Merise**, c'est-à-dire :

* Dictionnaire de données
* Matrice des dépendances fonctionnelles
* **M**odèles **C**onceptuels de **D**onnées
* **M**odèle **P**hysique de **D**onnées
* **M**odèle **L**ogique de **D**onnées

# Merise

Merise est une méthode d’organisation de base de données française et nous a permis de réaliser une base de données de manière optimale et donc par conséquent de simplifier nous requêtes SQL.

Cette méthode se décompose en **5 étapes** :

1. Dictionnaire de données 🡪 répertorie tous les champs de la base de données
2. Matrice des dépendances fonctionnelles 🡪 Permet de mettre en évidence les relations entre champs
3. Modèle Conceptuel de Données (**MCD**) 🡪 Logique de l’organisation des informations et de leurs relations
4. Modèle Logique de Données (**MLD**) 🡪 Traduction du MCD en y ajoutant les clés primaires et étrangères
5. Modèle Physique de Données (**MPD**) 🡪 Représentation graphique de la base de données

## Dictionnaire de données

Une image contenant texte

Description générée avec un niveau de confiance élevé

**Figure 7** : Dictionnaire de données

Ce tableau permet de répertorier les champs, leurs tailles, ainsi que leurs utilisations dans la base de données.

## Matrice des Dépendances Fonctionnelles

La Matrice des Dépendances Fonctionnelles permet de connaître les champs ayant des relations entres eux.

**Exemple :**

|  |
| --- |
| Individu |
| **ID\_individu**  Prenom  Nom  Age  Sexe |

À partir de l’**ID\_individu** j’obtiens le les autres informations de la table, ici Prenom, Nom, Age et Sexe.

Mais à l’inverse à partir de Prenom je ne peux rien obtenir car deux personnes peuvent avoir le même prénom.

**Figure 8** : Exemple table individu

Une image contenant capture d’écran

Description générée avec un niveau de confiance élevéNous avons donc créé une matrice à partir des champs du dictionnaire de données réalisé auparavant :

**Figure 9** : Matrice des Dépendances Fonctionnelles [partie 1]

|  |  |
| --- | --- |
|  | Dépendance impossible |
|  | Dépendance 🡪 avec **abscisse** on obtient **ordonnée(s)** |

**Figure 10** : Correspondance couleurs MDF

## Modèle Conceptuel de Données

**Figure** **11** : Matrice des Dépendances Fonctionnelles [partie 2]

Une image contenant texte, carte

Description générée avec un niveau de confiance très élevéLe modèle conceptuel de données ou **MCD** permet d’écrire de façon formelle les données qui seront par la suite utilisé dans le système d’informations. Il s’agit d’une représentation des données facilement compréhensible et permettant de décrire le système d’information à l’aide d’entités.

**Figure 12** : MCD Total

Une image contenant texte, carte

Description générée avec un niveau de confiance très élevé

**Figure 13** : Haut gauche MCD

Tout d’abord, on peut noter que les différentes tables tournent autour de la table **Vente**, cette table contient le champ *Date\_creation*, ce champ possède beaucoup de dépendances, c’es donc pourquoi cette table est au centre.

La table **Lieu** est liée à **Vente** grâce à l’entité Localiser, elle contient le champ *Province* ainsi que *ID\_province* dans laquelle on va chercher la province ainsi que si ID.

Elle est reliée à la table **Ville** grâce à l’entité Situer qui permettra donc d’obtenir les villes correspondant à la province.

La table **Lieu** est également reliée à la table **Guerre** via Avoir lieu pour permettre de savoir quand et ou ont eu lieu les différentes guerres.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table 1** | **Table 2** | **Cardinalité 1** | **Description** | **Cardinalité 2** | **Description** |
| Vente | Lieu | 1, 1 | Une vente n’a lieu qu’à un endroit | 0, n | Un endroit peut accueillir n ventes |
| Lieu | Ville | 1, 1 | Une ville ne se situe que dans une province | 0, n | Plusieurs villes peuvent avoir le même nom |
| Guerre | Lieu | 0, n | Il peut y avoir n guerre dans une ville | 1, 1 | Il ne peut n’y avoir qu’une guerre à la fois dans un lieu |

Une image contenant texte, carte

Description générée avec un niveau de confiance très élevé

**Figure 14** : Haut droite MCD

La table **Vente** est reliée à la table **Mois** grâce à l’entité Dater et permet d’obtenir les différents de vente.

Cette même table **Mois** est liée grâce à Correspondre à la table **Divinite** qui permet d’obtenir la divinité selon le mois.

La table **Divinite** est reliée à **Decoration** et **Pouvoir** via respectivement Représenter et Correspondre.

Puis la table **Vente** est lié à la table **Prix** via Valoir qui possède la *quantité*.

La table **Prix** possède la relation récursive Convertir en Ob qui comporte *Conversion* et *Monnaie*.

Cette relation va convertir la monnaie en *Oo*, *Oa*, *Of*, etc. En *Ob* (monnaie la plus « faible »).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table 1** | **Table 2** | **Cardinalité 1** | **Description** | **Cardinalité 2** | **Description** |
| Vente | Mois | 1, 1 | Une vente n’a lieu qu’à un moment donnée. | 0, n | Plusieurs ventes peuvent être simultanées |
| Mois | Divinite | 1, 1 | Un mois n’a qu’une divinité | 0, n | Une divinité n’a pas forcement de mois |
| Vente | Monnaie | 1, n | On peut payer avec plusieurs monnaies | 1, n | On peut acheter plusieurs choses avec la monnaie |
| Monnaie | Monnaie | 1, 1 | Une monnaie n’a qu’une conversion en Ob | 1, 1 | Des Ob ne correspondent qu’à une somme en une autre monnaie. |

Une image contenant texte, carte

Description générée avec un niveau de confiance très élevé

**Figure 15** : MCD Bas

Ici **Vente** est relié à 4 autres tables :

* **Objet**
* **Artisan**
* **Pouvoir**
* **Decoration**

Sa liaison avec **Decoration** se fait via l’entité Vendre et permet de savoir le nom de l’objet vendu 🡪 *Objet*

Sa seconde liaison se fait avec la table **Artisan** en utilisant l’entité Fabriquer.

Cela permet de connaître le nom de l’artisan qui a fabriqué objet 🡪 *Artisan*

La liaison avec pouvoir se fait grâce à Comporter et donne le nom du pouvoir que comporte l’objet 🡪 *Pouvoir*

Et pour finir **Vente** à une quatrième et dernière avec la table **Decoration** possible grâce à l’entité *Comporter*. Elle permet de connaître les décorations appliquées sur l’objet 🡪 *Nom\_deco*.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table 1** | **Table 2** | **Cardinalité 1** | **Description** | **Cardinalité 2** | **Description** |
| Vente | Objet | 1, 1 | On ne vent qu’un objet à la fois | 0, n | On peut vendre plusieurs fois le même modèle d’objet |
| Vente | Artisan | 1, n | On peut avoir 1 à n artisans qui vendent | 0, n | Un artisan peut vendre 0 à n objet |
| Vente | Pouvoir | 0, 1 | Un objet a ou d’a pas de pouvoir | 0, n | On peut vendre n fois un objet avec un pouvoir |
| Vente | Décoration | 0, n | Un objet peut avoir 0, 1, ou 2 décourations | 0, n | On peut vendre n fois un objet avec n pouvoir |

# Implémentation de la BDD

Pour implémenter la base de données de **Excel** vers **MySQL** nous avons utilisé la méthode du fichier CSV.

Pour ce faire on va réaliser un script SQL qui va transférer les données du fichier Excel vers un fichier CSV qui servira d’intermédiaire entre MySQL et Excel

On créer la base de données en **UTF-8** pour les caractères spéciaux. On créer les tables en faisant attention à ce que les clés étrangères ne prennent pas en compte une table pas encore créer.

Une fois que nous avons créé des tables pour importer les données à partir de fichiers CSV sous allons pouvoir insérer pour chaque champ à partir des tables d'importation. Puis nous pouvons enlever ces tables d'importation qui ne sont plus utiles. Enfin il faut UPDATE les clés étrangères afin de compléter les tables souhaitées.

# Requêtes

## Requête I : « Divinité préférée par province »

**Figure 16** : Requête I

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

SELECT permet de sélectionner

## Requête II : « Prix moyen des équipement »

**Figure 17** : Requête II

**SELECT** Objet permet comme son nom l’indique de sélectionner la table **Objet**

Puis grâce à **avg**(Oo/Oa/Of) \* le multiple de conversion vers Ob ; on finit par diviser par la quantité et on obtient le **prix moyen d’un équipement en Ob**.

**FROM Vente** permet de savoir dans quelle table chercher les informations, ici la table **Vente.**

**Group By objet** permet de grouper les résultats de la requête.

## Une image contenant capture d’écran Description générée avec un niveau de confiance très élevéRequête III : « Paiement en Oo ou Oa par province »

**Figure 18** : Requête III

**SELECT** lieux.nom\_prov permet de sélectionner le champ *nom\_prov* de la table **Lieu**

**From** vente permet de diriger la requête vers la table **Vente**.

**Inner join** permet de lier *ID\_lieu*es des tables **Lieux** et **Vente**

**Group By** Puis on groupe les résultats de la requête

## Une image contenant capture d’écran Description générée avec un niveau de confiance élevéRequête IV : « Comparaison ventes provinces & Egypte »

**Figure 19** : Requête IV

**SELECT LEFT** permet de sélectionner le troisième caractère de la date (cela donnera l’année).

**COUNT** va ici compter quand la vente provient de la province « **AEgypte** »

Puis un second **COUNT** va compter les ventes qui ont lieu dans les provinces autres que « **AEgypte** »

**INNER JOIN** permet de lier les tables **Vente** et **Lieu**

Enfin, **GROUP BY** va organiser les résultats en sortie de la requête

## Une image contenant capture d’écran Description générée avec un niveau de confiance très élevéRequête V : «  Comparaison ventes avec et sans fête religieuse »

**Figure 20** : Requête V

**SELECT** va permettre de sélectionner la table (ici **Mois**) et le champ *Nom\_divinite*.

**COUNT** va compter le nombre de vente selon le mois.

**INNER JOIN** va lier les champs *Mois* dans les deux tables **Vente** et **Mois**.

Puis **GROUP BY** va permettre de regrouper les résultats selon la divinité du mois.

Enfin, **ORDER BY** va tri les valeurs par ordre décroissant.

## Une image contenant capture d’écran Description générée avec un niveau de confiance très élevéRequête VI : « Comparaison ventes avec et sans guerre »

**Figure 21** : Requête VI

**SELECT LEFT** permet de sélectionner le troisième caractère de la date (cela donnera l’année) **FROM** (depuis) la table **Vente**.

**INNER JOIN** va lier les tables **Vente** et **Guerre** via les champs *Date\_creation* et *annee\_guerre*.

Le second **INNER JOIN** va lier **Vente** et **Lieu** par le champ *ID\_province*.

**GROUP BY** va regrouper les provinces et les années de guerre puis le second **s** va regrouper les provinces et leurs nombres de ventes.

## Requête VII : « Prix de vente artisan selon la divinité »

Nous n’avons pas réussi à réaliser, elle sera donc dans les points à améliorer de notre diaporama.

# Procédure

Le but de cette procédure est de permettre à Héphaïstos de rentrer automatiquement les nouvelles ventes dans la base de données MySQL.

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

**Figure 22** : Procédure stockée

**DROP PROCEDURE IF EXISTS** Permet de créer la procédure si elle n’existe pas.

**DELIMITER |** Change le délimiter actuel pour « | »

**CREATE PROCEDURE** (paramètres de la procédure) permet de créer la procédure ainsi qu’ajouter les paramètres.

**BEGIN** et **END** permettent de délimiter une suite d’instruction SQL.

**INSERT INTO ()** permet d’insérer dans des champs les valeurs de **VALUES ()**

# Une image contenant capture d’écran Description générée avec un niveau de confiance très élevéVue

**Figure 23** : Vue simplifiée

**CREATE VIEW** permet de créer une vue

**SELECT** permet de sélectionner les artisans dans la table **Artisan**, on va grâce à **count** compter le nombre de ventes dans la table **Vente** et ce tous les ans dans le champ *Annee\_creation* de la table **Vente**.

**FROM** montre qu’on va dans la table **Artisan**.

**INNER JOIN** artisan permet de lier deux tables entre elles, ici les tables **Artisan** et **Vente**.

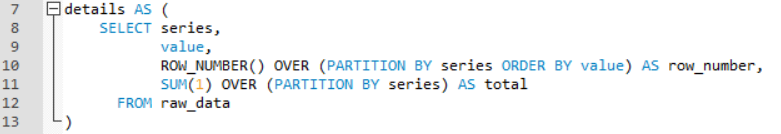
**WHERE** permet de choisir l’artisan dont on veut observer l’évolution des ventes dans le temps.

**GROUP BY** dit à la requête de trier les années.

# Une image contenant capture d’écran Description générée avec un niveau de confiance très élevéBoîte à moustache

**Figure 24** : Boîte à moustache 1

Cette partie va permettre d’initialiser la boîte à moustache pour la table **Vente**.



**Figure 25** : Boîte à moustache 2

Cette partie va permettre de compter les données et de les ordonner depuis la table **Vente**.

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

**Figure 26** : Boîte à moustache 3

Cette partie va faire les différents calculs, tout d’abord le premier quartile, puis le troisième quartile.

Dans cette dernière partie, le script SQL va réaliser le minimum, la médiane ainsi que le maximum.

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

**Figure 27** : Boîte à moustache 4

# Bilan personnel

### Valérian POIRIER

#### Investissement et travail fait

Durant ce projet, j’ai été chargé de la création des différentes requêtes.

L’implémentation de la base de données n’ayant pas été finie, je n’ai pas pu tester l’intégralité de mes requêtes, j’ai pu en tester certaines en créant de « fausses » valeurs manuellement. Mais je n’ai pas pu le faire pour toutes les requêtes.

#### Impression sur la semaine

J’ai trouvé que le sujet n’était pas tout le temps clair, au niveau des requêtes demandées, il n’y avait pas assez de précision sur ce que l’on devait faire. Ça a aussi mené à un problème de compréhension chez les tuteurs, qui n’étaient pas tous d’accords sur les mêmes points.

Malgré tout, j’ai apprécié le projet, la manipulation de la base de données à l’aide de requêtes est quelque chose que je trouve assez intéressant, et le contexte du sujet était aussi intéressant.

Pour ma part, j’aurai pu améliorer certains points, j’ai toujours tendance à rester seul dans mes problèmes, malgré des améliorations par rapport aux projets précédents à ce niveau, je manque encore de communication. J’aurai pu aussi aider pour l’implémentation de la base de données, ayant été bloqué à cause de ça, mais je n’y ai rien fait non plus.

#### Impression sur le groupe

Je n'ai pas eu de soucis particuliers avec le groupe, tout le monde travaillait et le chef de projet venait souvent voir l'avancement de chacun pour voir comment ça se passait.

### Simon MANIEZ

#### Investissement et travail fait

Je me suis occupé de l’implémentation de la base de données tout au long du projet ce fut une partie intéressante et compliquée.

#### Impression sur la semaine

Un projet intéressant, une compréhension du sujet compliquer.

#### Impression sur le groupe

Un bon groupe, aucun souci particulier pour ma part. Un bon chef de projet.

Une répartition des tâches qui aurait été plus facile à quatre.

### Paul FICOT

#### Investissement et travail fait

J’ai réalisé l’intégralité de l’organisation (Gantt, RACI, WBS, etc.) ainsi que la modélisation de la base données avec la méthode Merise (Dictionnaire de données, Matrice des dépendances fonctionnelles, MCD, MLD, MPD).

J’ai également réalisé la boîte à moustache.

#### Impression sur la semaine

Cette semaine fut intense mais qui s’est tout de même assez bien passé dans l’ensemble malgré quelques problèmes d’implémentation et aucune réponse pour la validation de notre MCD.

#### Impression sur le groupe

Je suis plutôt satisfait de notre groupe, tout le monde a travaillé de manière constante.