

SAE 1.04 : Création d'une base de données

BUT Informatique – 1A

Alexandre CHENAL,

Kajan ARULNATHAN,

Mikael GALTIER,

Ariles DADI,

Ryan BACHATENE,

Zayd CROMBEZ.

Table des matières :

- 1) Introduction
- 2) Développement Détaillé
- 3) Implémentation sur Dbeaver
- 4) Tests
- 5) Difficultés rencontrés
- 6) Conclusion

1/Introduction :

Contexte :

En tant qu'élèves en première année de BUT Informatique, il nous a été demandé de créer une base de données SQLite à partir de données disponibles sur le site de l'Assurance Maladie Data Ameli. L'objectif est d'unifier et de consolider 3 types de données séparées en une seule base de données afin de faciliter la navigation et la visualisation des données. Cette saé a pour but de consolider et d'évaluer nos connaissances en SQL mais aussi notre capacité à découvrir et à s'appropriier de nouveaux logiciels tels que DBeaver. La saé 1.04 est la première partie d'un projet plus large aboutissant à l'implémentation d'un besoin client via la création d'une application web qui fera l'objet de la saé 1.01. Ce rapport va donc détailler notre cheminement de pensées ainsi que les difficultés que nous avons rencontrées.

2/Développement détaillé :

Répartition des tâches :

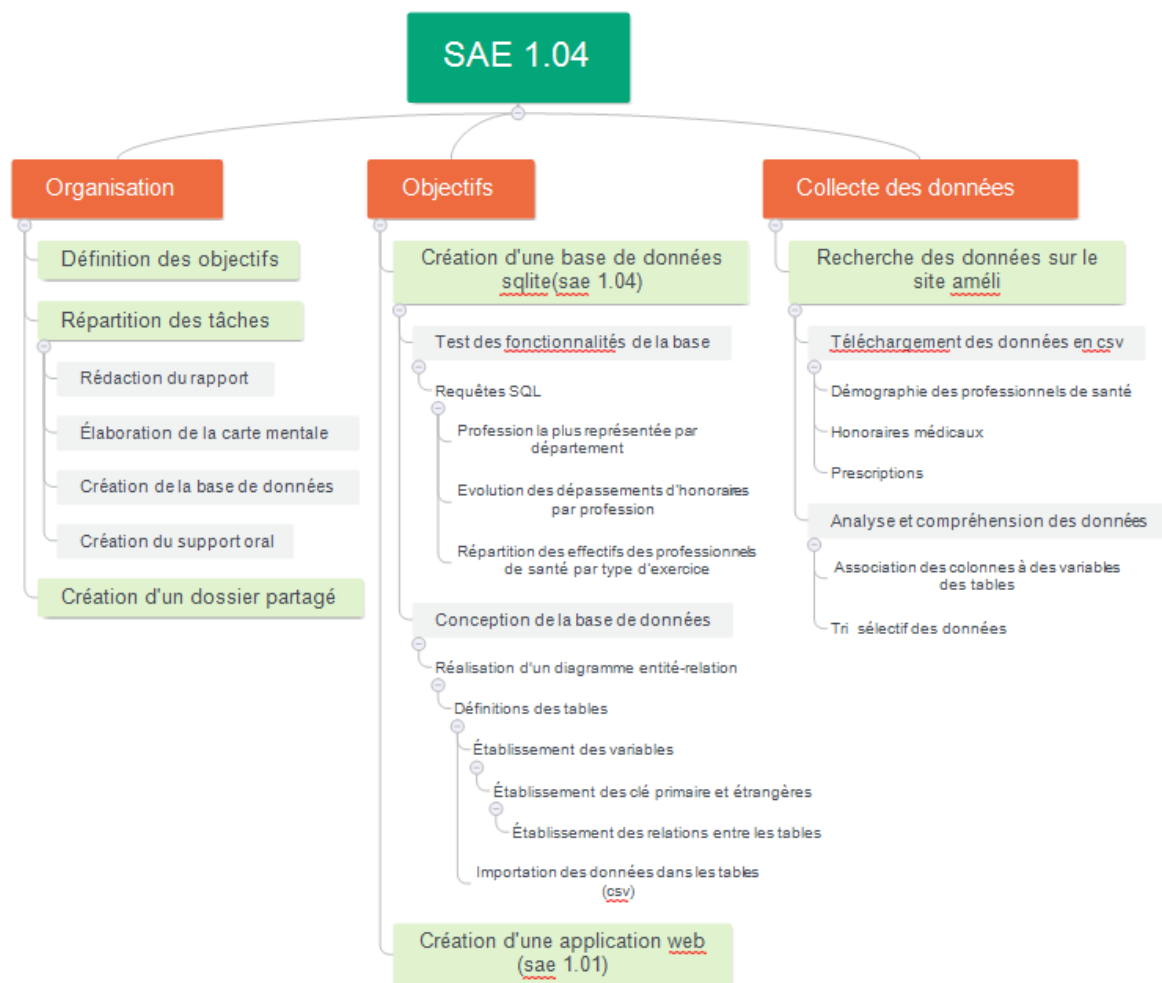
Dans cette saé, nous sommes 6 étudiants en première année de BUT informatique pour peu de tâches différentes à réaliser, il était donc crucial de bien répartir les tâches pour optimiser pleinement notre utilisation des séances d'autonomie. Nous nous sommes alors penchés avant toute chose sur cette question. Tout d'abord nous avons choisis notre chef de projet. Nous avons sélectionné Kajan ARULNATHAN sans trop de discussions car il a déjà été chef de projet dans les saé précédentes et il connaît donc bien son rôle. Cela étant fait, nous nous sommes donc tournés vers la question de la répartition des tâches. Pour le début, nous nous sommes divisés en deux groupes. Un qui se charge d'aller récupérer les données sous format CSV (pour une meilleure lisibilité et pour une importation plus simple) et l'autre qui avait pour tâche de réaliser la

carte mentale. Une fois ces deux tâches réalisées, nous nous sommes réunis pour analyser les données.

Réalisation de la carte mentale :

Pour avoir une vision plus claire du projet et pour définir clairement les objectifs, nous nous sommes lancés dans la réalisation d'une carte mentale sur le logiciel MindView.

Voici le résultat final ci-dessous :



C'est donc à partir de cette carte mentale que nous nous sommes appuyés pour le reste de la saé.

Collecte et Analyse des données :

Avant de créer une base de données, il fallait ... des données ! Conformément au cahier des charges nous nous sommes alors rendus sur le site de l'Assurance Maladie : Data Ameli. Nous avons donc téléchargé les trois fichiers CSV suivants :



Data professionnels de santé libéraux

Explorer de manière interactive les données relatives aux professionnels de santé libéraux.

[Accéder à Data professionnels de santé libéraux](#) →



Professionnels de santé libéraux : effectif par type d'exercice libéral et par territoire (département, région)

Ce jeu de données est une des sources de datavisualisations disponibles sur le site Data professionnels de santé libéraux.

| | |
|-------------------|--|
| Producteur | Caisse nationale de l'Assurance Maladie (Cnam) |
| Licence | Open Database License (ODbL) |

 Tableau
 Export
 API



Professionnels de santé libéraux : montants des prescriptions par poste et par territoire (département, région)

Ce jeu de données est une des sources de datavisualisations disponibles sur le site Data professionnels de santé libéraux.

| | |
|-------------------|--|
| Producteur | Caisse nationale de l'Assurance Maladie (Cnam) |
| Licence | Open Database License (ODbL) |

 Tableau
 Export
 API



Professionnels de santé libéraux : montants des honoraires par catégorie et par territoire (département, région)

| | |
|-------------------|--|
| Producteur | Caisse nationale de l'Assurance Maladie (Cnam) |
| Licence | Open Database License (ODbL) |

 Tableau
 Analyse
 Export
 API

3/Implémentation sur Dbeaver

Après avoir analysé les données nous avons donc procédé à l'implémentation sur Dbeaver. Nous avons d'abord créé les tables indépendantes. L'objectif était de définir les tables qui ne dépendent d'aucune autre table pour leur structure ou leur contenu. Les tables créées dans cette phase incluent Regions, avec les colonnes region (clé primaire) et libelle_region, Départements, contenant departement (clé primaire), libelle_departement, et une clé étrangère région référencée à la table Regions, et ProfessionsSante, qui contient une seule colonne profession_sante (clé primaire) Les relations entre les tables ont été intégrées grâce à l'ajout de clés étrangères. Honoraires a été créée avec des colonnes telles que id_honoraires (clé primaire), annee, total_honoraires, departement (référéncé à Départements), et profession_sante (référéncé à ProfessionsSante). D'autres tables liées aux données spécifiques, comme Prescription ou Démographie, ont également été implémentées en suivant ce même principe. (cf. Annexe 1)

Étapes d'implémentation des données :

1. Sélection de la Table dans DBeaver : Une table a été sélectionnée dans DBeaver en effectuant un clic droit pour accéder à l'option d'importation des données.
2. Choix du Fichier CSV : Le fichier CSV correspondant aux données à intégrer a été sélectionné via l'interface d'importation. DBeaver offre la possibilité de spécifier le chemin du fichier et de configurer son format (par exemple, séparateur de colonnes)
3. Vérification des Colonnes : Une vérification des colonnes a été réalisée pour s'assurer que leur structure et leurs types correspondent à la table définie. Si nécessaire, des ajustements ont été faits pour aligner les colonnes du fichier CSV avec celles de la table 2024-2025 | IUT Créteil-Vitry | BUT Informatique 5
4. Finalisation de l'Importation : Une fois les configurations validées, l'importation des données dans la table SQLite a été finalisée en lançant le processus. Les données transférées ont été vérifiées directement dans la table cible.

Test

Les données présentes dans la base ont été vérifiées pour s'assurer que les professions de santé, les régions et les départements sont correctement enregistrés. Nous avons également validé la disponibilité des années pour les données d'honoraires et de prescriptions.

Des analyses ont été réalisées pour calculer les prescriptions totales par région et identifier les régions les plus dépensières. Nous avons également analysé les dépassements moyens d'honoraires par département afin de relever les départements ayant les valeurs les plus élevées.

Une comparaison des données a été effectuée pour examiner les prescriptions moyennes entre régions et professions. Nous avons aussi étudié l'évolution des dépenses régionales au fil des années ainsi que les variations des dépassements moyens en fonction des professions et des années.

Enfin, une répartition des professionnels a été réalisée en comptabilisant leur nombre selon leur type d'exercice, qu'ils soient généralistes ou spécialistes.

```
-- Evolution des dépassements d'honoraires par profession.
SELECT ProfessionsSante.profession_nom, Honoraires.annee, SUM(Honoraires.montant_moyen_honoraires) AS depassements_honoraires
FROM Honoraires
JOIN ProfessionsSante ON Honoraires.profession_id = ProfessionsSante.profession_id
GROUP BY ProfessionsSante.profession_nom, Honoraires.annee
ORDER BY ProfessionsSante.profession_nom, Honoraires.annee
LIMIT 20;
```

```
-- Profession la plus représentée par département.
SELECT Departements.departement_nom, ProfessionsSante.profession_nom, MAX(Demographie.effectif) AS max_effectif
FROM Demographie
JOIN ProfessionsSante ON Demographie.profession_id = ProfessionsSante.profession_id
JOIN Departements ON Demographie.departement_id = Departements.departement_id
GROUP BY Departements.departement_nom, ProfessionsSante.profession_nom
ORDER BY max_effectif DESC
LIMIT 20;
```

```
-- Répartition des effectifs des professionnels de santé par type d'exercice.
SELECT Demographie.libelle_type_exercice_liberal, ProfessionsSante.profession_nom, SUM(Demographie.effectif) AS total_effectifs
FROM Demographie
JOIN ProfessionsSante ON Demographie.profession_id = ProfessionsSante.profession_id
GROUP BY Demographie.libelle_type_exercice_liberal, ProfessionsSante.profession_nom
ORDER BY Demographie.libelle_type_exercice_liberal, ProfessionsSante.profession_nom
LIMIT 20;
```

Difficultés rencontrées

Cette partie présente les principales difficultés rencontrées au cours de la création de la base de données.

L'utilisation de DBeaver, un nouveau logiciel pour l'équipe, a constitué une difficulté majeure. L'apprentissage de ses fonctionnalités et de son interface a nécessité un temps d'adaptation. Cette phase d'apprentissage a ralenti le démarrage du projet.

Des problèmes ont également été rencontrés lors de l'implémentation de la base de données. La configuration des relations, la définition des clés primaires et étrangères, ainsi que l'ajustement des types de données se sont révélés plus complexes que prévu.

La manipulation de fichiers volumineux également a posé des difficultés d'analyse et d'importation. Certains fichiers contenaient un grand nombre de colonnes ou de lignes, rendant leur traitement difficile.

Conclusion

Dans le cadre de la SAE 1.04, nous avons conçu une base de données relationnelle avec SQLite pour organiser et analyser les données de santé libérale. Après avoir étudié les fichiers CSV et créé les tables, nous avons manipulé les données et utilisé des outils comme DBeaver pour faciliter le processus. Malgré quelques défis rencontrés, l'objectif a été atteint, et la base est fonctionnelle et prête à être exploitée pour la suite du projet.

2024-2025 | IUT Créteil-Vitry | BUT Informatique 8 Annexe Liste des annexes : ● Annexe 1 : Requêtes SQLite 2024-2025 | IUT Créteil-Vitry | BUT Informatique 9 Annexe 1 : Requêtes SQLite 2024-2025 | IUT Créteil-Vitry | BUT Informatique 1

Base de Données final :

