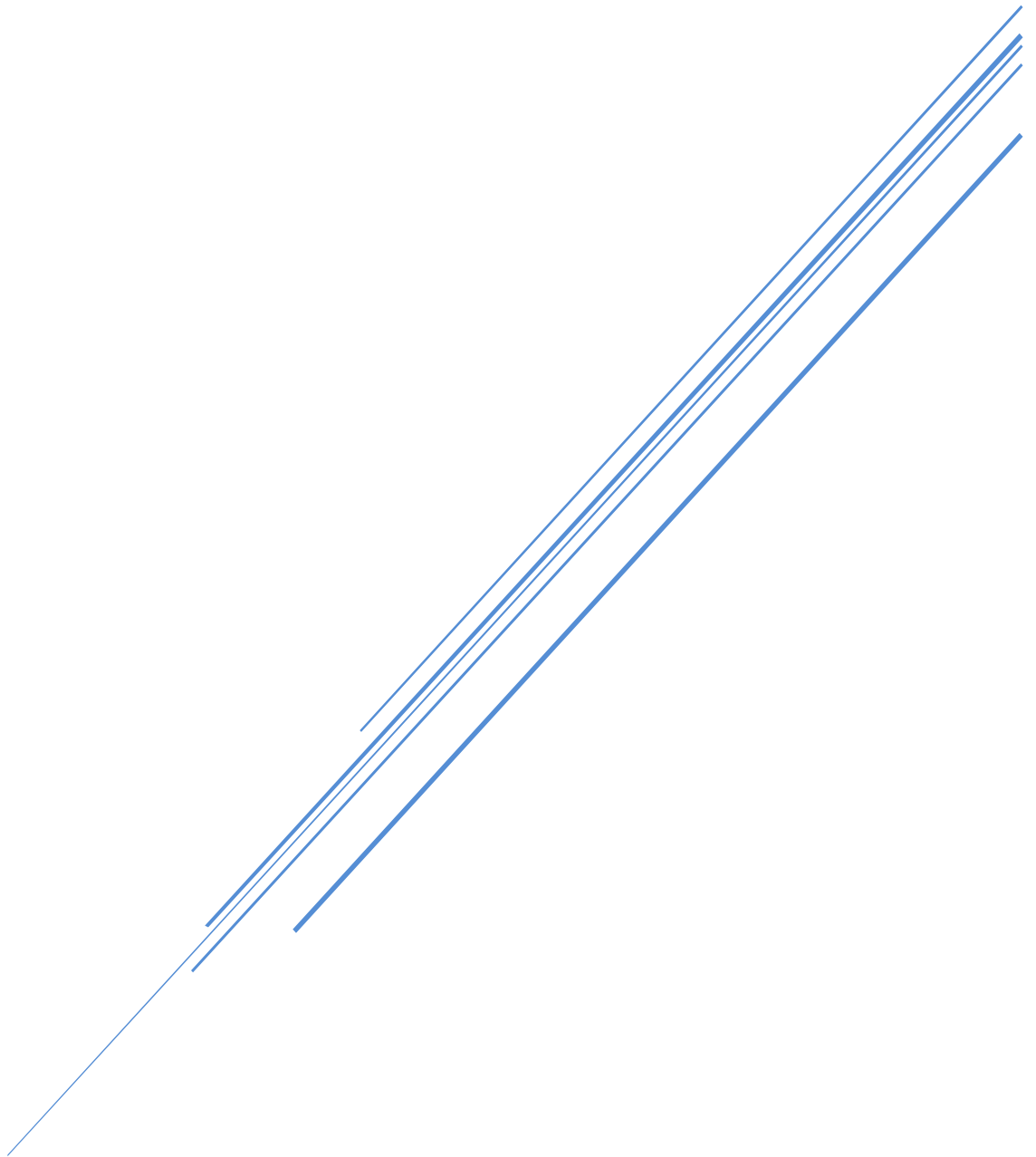


# MODÉLISATION CONCEPTUELLE

Groupe 4 – Marengo – Friedli - Marzullo



BDR

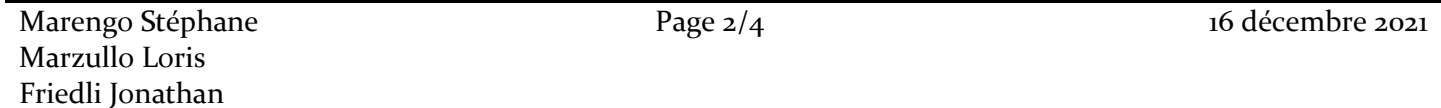
## Table des matières

1	Introduction .....	2
2	Modèle relationnel .....	2
2.1	Représentation graphique avec DBeaver .....	2
2.2	Contraintes d'intégrité .....	3
3	Script SQL .....	3
4	Conclusion .....	4

Dans le cadre de ce projet, nommé Concer'ino, nous devons réaliser une application complète de base de données relationnelle. Ce projet est à réaliser en groupe de trois personnes. Notre projet consistera en une application permettant de lister différents concerts dans toute la Suisse avec la possibilité d'indiquer sa présence.

Le présent document définit le modèle relationnel dudit projet ainsi que les contraintes d'intégrité dont il faudra tenir compte ultérieurement dans le script SQL car ne pouvant, pour l'heure, être exprimées dans les instructions CREATE.

## 2.1 Représentation graphique avec DBeaver



## 2.2 Contraintes d'intégrité

- Plusieurs concerts ne peuvent pas avoir lieu en même temps et au même endroit.
- Le nombre de personne inscrite à un concert ne peut pas excéder la capacité de sa salle.
- Un artiste ne peut pas se produire lors de plusieurs concerts en même temps.
- Un utilisateur ne peut pas s'inscrire à différents concerts ayant lieu en même temps.
- Pour un même concert, 2 artistes ne peuvent pas avoir le même numéro de passage.
- Le numéro de passage est une suite (commençant à 1) continue de valeurs.
- Un utilisateur ne peut pas s'inscrire à un concert dont la date est déjà passée.
- Un concert ne peut être noté que par un utilisateur ayant assisté audit concert.
- Une salle ne peut être notée que par un utilisateur ayant assisté à au moins un concert dans cette dernière.
- Un artiste ne peut être noté que par un utilisateur ayant effectivement assisté à une de ses performances.
- Un groupe sans membres actifs ne peut pas se produire en concert.
- Un artiste solo ne peut pas rejoindre un groupe dont il fait déjà partie.

## 3 Script SQL

Cette partie développe certains de nos choix lors de l'écriture du script SQL

Afin de rendre notre modèle plus explicite, nous avons nommé les relations définissant les notes « NoteX » en lieu et place de la convention de nommage « Utilisateur\_X » (où X peut être Lieu, Concert ou Artiste).

Pour cette même raison, la relation caractérisant les artistes solos faisant partie d'un groupe est appelée Membre, à défaut d'être appelée ArtisteSolo\_Groupe.

Dans la table Utilisateur, nous avons choisi le type CHAR(60) pour l'attribut motDePasse car celui-ci contiendra un mot de passe hashé.

## 4 Conclusion

Toutes les contraintes triviales ont été vérifiées à l'aide de l'instruction CHECK.

Nous avons remarqué que la plupart des contraintes d'intégrité que nous avons ne peuvent pas être gérées pour le moment mais devront l'être ultérieurement.

Globalement, nous pensons que l'état actuel de notre base de données nous permet de continuer ce projet sans accroc.