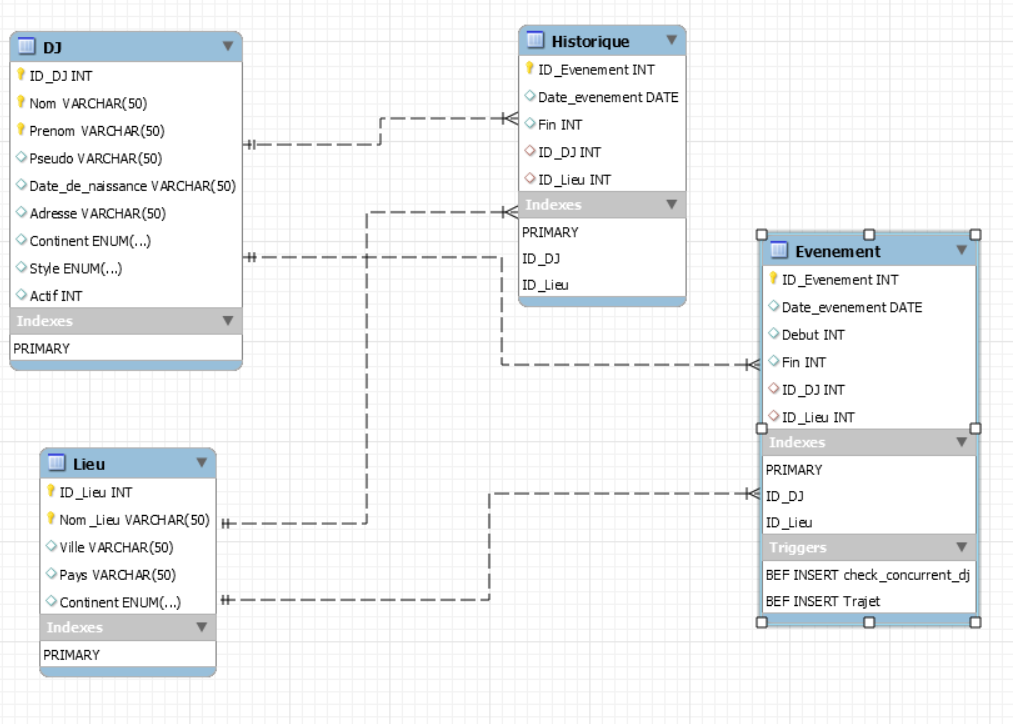
**Rapport technique**

Pour mener à bien le projet, nous avons commencé par initialiser la base de données à l’aide de MySQL Workbench :



Nous avons donc créé 4 tables :

* DJ : pour recevoir les informations sur les DJ
* Lieu : regroupant tous les lieux organisant des évènements
* Évènement : regroupant tous les évènements à venir
* Historique : regroupant tous les évènements passés

Ainsi que 2 triggers :

* Check\_concurent\_dj : qui vérifie qu’un évènement ne se superpose avec un autre sur un même lieu
* Trajet : pour permettre aux DJ d’avoir le temps de se déplacer entre 2 évènements

En plus d’Event qui actualise la table Historique de tous les évènements qui se sont passés dans la journée.

Pour les procédures on a :

- ajout\_DJ(nom varchar(50),prenom varchar(50),pseudo varchar(50),date\_naissance date, adresse varchar(50), continent varchar(50),style varchar(50),actif int) : qui rajoute un DJ à la table DJ avec un id auto-incrémenté

- create procedure supp\_DJ(id int) : qui modifie un DJ pour passer sa valeur actif à 0, cela nous permet de conserver sans problème tous les évènements liés à ce DJ.

- create procedure modif\_DJ(id int, nomDJ varchar(50),prenomDJ varchar(50),pseudoDJ varchar(50),date\_naissance date, adresseDJ varchar(50), continentDJ varchar(50),styleDJ varchar(50),actifDJ int) : afin de modifier certains paramètres d’un DJ de la table.

- ajout\_evenement(IN date\_event DATE, IN debut INT, IN fin INT, IN Pseudo\_dj VARCHAR(50), IN lieu\_nom VARCHAR(50)) : afin d’ajouter un évènement sur la table Évènement à l’aide du pseudo du DJ et du nom du lieu.

- EvenementMois(mois int) : afin d’avoir la liste des évènements se déroulant un mois donné.

- Top5() : pour avoir le Top 5 des DJ ayant participé au plus d’évènement.

Ensuite nous avons fait le lien entre notre application web et cette base de données grâce à un front-end en css et le back-end en JavaScript en utilisant l’API Rest afin de lier ces 3 parties.