

MLD :

Medicaments = (id INT, nom VARCHAR(255), date_creation DATE, laboratoire_createur VARCHAR(255), dose_max_journaliere VARCHAR(50));

Liste_effets_therapeutiques = (id INT, effet TEXT);

Liste_effets_secondaires = (id INT, effet TEXT);

Activites = (id INT, nom VARCHAR(255), description TEXT, date_heure DATETIME);

Utilisateurs = (email VARCHAR(255), nom VARCHAR(255), prenom VARCHAR(255));

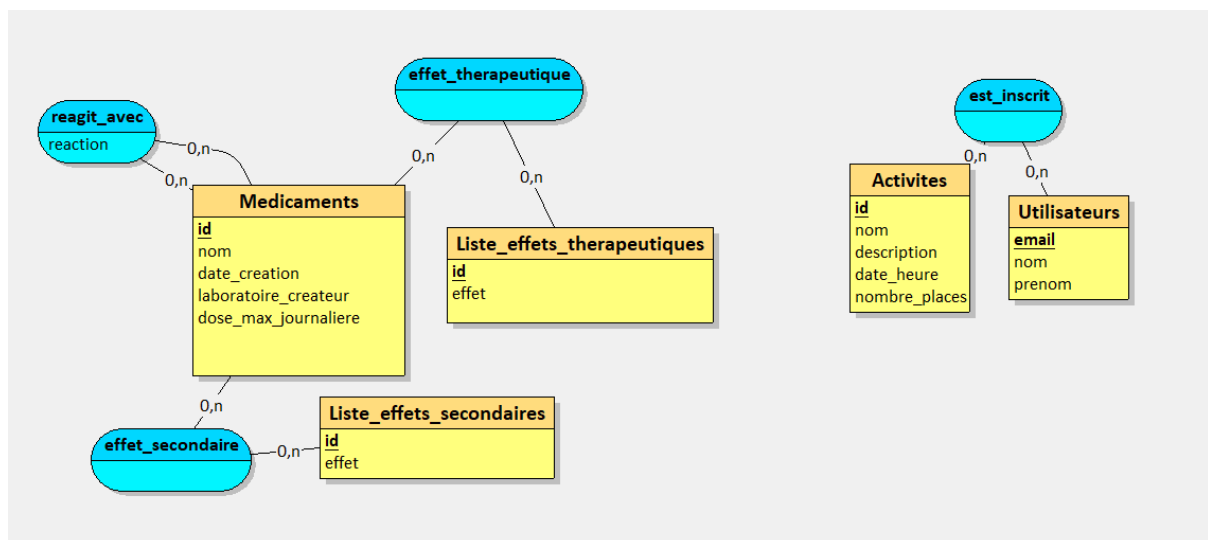
reagit_avec = (#id_medicament_1, #id_medicament_2, reaction TEXT);

effet_therapeutique = (#id_medicament, #id_effet);

effet_secondaire = (#id_medicament, #id_effet);

est_inscrit = (#id_activite, #id_utilisateur);

MCD



Script SQL :

```
CREATE TABLE Medicaments(  
  id INT,  
  nom VARCHAR(255),  
  date_creation DATE,  
  laboratoire_createur VARCHAR(255),  
  dose_max_journaliere VARCHAR(50),  
  PRIMARY KEY(id)  
);  
  
CREATE TABLE Liste_effets_therapeutiques(  
  id INT,  
  effet TEXT,  
  PRIMARY KEY(id)  
);  
  
CREATE TABLE Liste_effets_secondaires(  
  id INT,  
  effet TEXT,  
  PRIMARY KEY(id)  
);  
  
CREATE TABLE Activites(  
  id INT,  
  nom VARCHAR(255),  
  description TEXT,  
  date_heure DATETIME,  
  nombre_places INT,  
  PRIMARY KEY(id)  
);  
  
CREATE TABLE Utilisateurs(  
  email VARCHAR(255),  
  nom VARCHAR(255),  
  prenom VARCHAR(255),  
  PRIMARY KEY(email)  
);  
  
CREATE TABLE reagit_avec(  
  id INT,  
  id_1 INT,  
  reaction TEXT,  
  PRIMARY KEY(id, id_1),  
  FOREIGN KEY(id) REFERENCES Medicaments(id),  
  FOREIGN KEY(id_1) REFERENCES Medicaments(id)  
);  
  
CREATE TABLE effet_therapeutique(  
  id INT,
```

```

id_1 INT,
PRIMARY KEY(id, id_1),
FOREIGN KEY(id) REFERENCES Medicaments(id),
FOREIGN KEY(id_1) REFERENCES Liste_effets_therapeutiques(id)
);

```

```

CREATE TABLE effet_secondaire(
id INT,
id_1 INT,
PRIMARY KEY(id, id_1),
FOREIGN KEY(id) REFERENCES Medicaments(id),
FOREIGN KEY(id_1) REFERENCES Liste_effets_secondaires(id)
);

```

```

CREATE TABLE est_inscrit(
id INT,
email VARCHAR(255),
PRIMARY KEY(id, email),
FOREIGN KEY(id) REFERENCES Activites(id),
FOREIGN KEY(email) REFERENCES Utilisateurs(email)
);

```

Triggers :

Rajouter une place lorsqu'un utilisateur de désinscrit :

```

CREATE TRIGGER `desinscription_activite` AFTER DELETE ON `est_inscrit` FOR EACH ROW
BEGIN
    UPDATE Activites
    SET nombre_places = nombre_places + 1
    WHERE id = OLD.id_activite;
END

```

Enlever une place lorsqu'un utilisateur s'inscrit

```

CREATE TRIGGER `inscription_activite` AFTER INSERT ON `est_inscrit` FOR EACH ROW BEGIN
    UPDATE Activites
    SET nombre_places = nombre_places - 1
    WHERE id = NEW.id_activite;
END

```

Supprimer toutes les inscriptions d'un utilisateur lorsqu'un est supprimé de la table utilisateurs

```

CREATE TRIGGER `suppression_utilisateur` AFTER DELETE ON `utilisateurs` FOR EACH ROW
BEGIN
    DELETE FROM est_inscrit WHERE email = OLD.email;
END

```