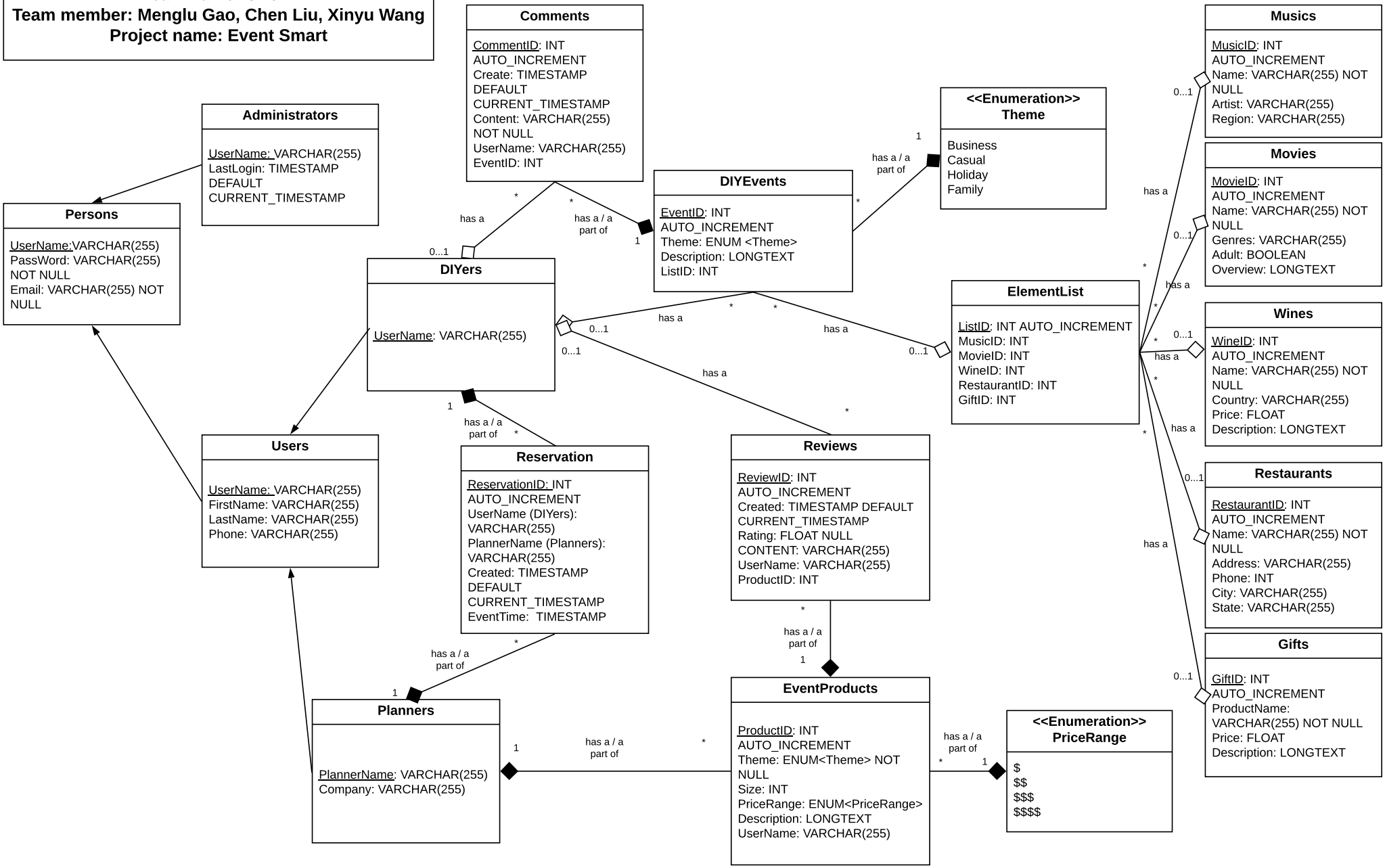
Team name: GLoW
Team member: Menglu Gao, Chen Liu, Xinyu Wang
Project name: Event Smart



```
#Sql code
#TeamName: GLoW
#ProjectName: Event Smart
CREATE SCHEMA IF NOT EXISTS EventSmart;
USE EventSmart;

DROP TABLE IF EXISTS Comments;
DROP TABLE IF EXISTS Reviews;
DROP TABLE IF EXISTS Reservations;
DROP TABLE IF EXISTS DIYEvents;
DROP TABLE IF EXISTS EventProducts;
DROP TABLE IF EXISTS ElementLists;
DROP TABLE IF EXISTS Movies;
DROP TABLE IF EXISTS Musics;
DROP TABLE IF EXISTS Wines;
DROP TABLE IF EXISTS Restaurants;
DROP TABLE IF EXISTS Gifts;
DROP TABLE IF EXISTS DIYers;
DROP TABLE IF EXISTS Planners;
DROP TABLE IF EXISTS Users;
DROP TABLE IF EXISTS Administrators;
DROP TABLE IF EXISTS Persons;

CREATE TABLE Persons (
        UserName VARCHAR(255),
  Psw VARCHAR(255),
  Email VARCHAR(255),
  CONSTRAINT pk_Persons_UserName PRIMARY KEY (UserName)
);

CREATE TABLE Administrators (
        UserName VARCHAR(255),
  LastLogin TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  CONSTRAINT pk_Administrators_UserName PRIMARY KEY (UserName),
  CONSTRAINT fk_Administrators_UserName
                FOREIGN KEY (UserName)
    REFERENCES Persons(UserName)
    ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE Users (
        UserName VARCHAR(255),
  FirstName VARCHAR(255),
  LastName VARCHAR(255),
```

```
   Phone VARCHAR(255) DEFAULT NULL,
   CONSTRAINT pk_Users_UserName PRIMARY KEY (UserName),
   CONSTRAINT fk_Users_UserName
                    FOREIGN KEY (UserName)
      REFERENCES Persons(UserName)
      ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE DIYers (
          UserName VARCHAR(255),
   CONSTRAINT pk_DIYers_UserName PRIMARY KEY (UserName),
   CONSTRAINT fk_DIYers_UserName
                    FOREIGN KEY (UserName)
      REFERENCES Users(UserName)
      ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE Planners (
          PlannerName VARCHAR(255),
          COMPANY VARCHAR(255),
   CONSTRAINT pk_Planners_PlannerName PRIMARY KEY (PlannerName),
   CONSTRAINT fk_Planners_PlannerName
                    FOREIGN KEY (PlannerName)
      REFERENCES Users(UserName)
      ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE Musics (
          MusicID INT AUTO_INCREMENT,
   MusicName VARCHAR(255) NOT NULL,
   Artist VARCHAR(255),
   Genres VARCHAR(255),
   CONSTRAINT pk_Musics_MusicID PRIMARY KEY (MusicID)
);

CREATE TABLE Movies (
          MovieID INT AUTO_INCREMENT,
   MovieName VARCHAR(255) NOT NULL,
   DirectorName VARCHAR(255),
   Genres VARCHAR(255),
   ContentRating VARCHAR(255),
   CONSTRAINT pk_Movies_MovieID PRIMARY KEY (MovieID)
);
```

```
CREATE TABLE Wines (
        WineID INT AUTO_INCREMENT,
  WineName VARCHAR(255) NOT NULL,
  Country VARCHAR(255),
  Description LONGTEXT,
  Price INT,
  CONSTRAINT pk_Wines_WineID PRIMARY KEY (WineID)
);

CREATE TABLE Restaurants (
        RestaurantID INT AUTO_INCREMENT,
  RestaurantName VARCHAR(255) NOT NULL,
  Phone VARCHAR(255),
  Address TEXT,
  City VARCHAR(255),
  State VARCHAR(255),
  CONSTRAINT pk_Restaurants_RestaurantID PRIMARY KEY (RestaurantID)
);

CREATE TABLE Gifts (
        GiftID INT AUTO_INCREMENT,
  ProductName TEXT NOT NULL,
  Price FLOAT (6, 2),
  Category VARCHAR(255),
  Description LONGTEXT,
  CONSTRAINT pk_Gifts_GiftID PRIMARY KEY (GiftID)
);

CREATE TABLE ElementLists (
        ListID INT AUTO_INCREMENT,
  MusicID INT,
  MovieID INT,
  WineID INT,
  RestaurantID INT,
  GiftID INT,
  CONSTRAINT pk_ElementLists_ListID PRIMARY KEY (ListID),
  CONSTRAINT fk_ElementLists_MusicID
                FOREIGN KEY (MusicID)
    REFERENCES Musics (MusicID)
    ON UPDATE CASCADE ON DELETE SET NULL,
        CONSTRAINT fk_ElementLists_MovieID
                FOREIGN KEY (MovieID)
    REFERENCES Movies (MovieID)
    ON UPDATE CASCADE ON DELETE SET NULL,
```

```
        CONSTRAINT fk_ElementLists_WineID
                FOREIGN KEY (WineID)
    REFERENCES Wines (WineID)
    ON UPDATE CASCADE ON DELETE SET NULL,
        CONSTRAINT fk_ElementLists_RestaurantID
                FOREIGN KEY (RestaurantID)
    REFERENCES Restaurants (RestaurantID)
    ON UPDATE CASCADE ON DELETE SET NULL,
        CONSTRAINT fk_ElementLists_GiftID
                FOREIGN KEY (GiftID)
    REFERENCES Gifts (GiftID)
    ON UPDATE CASCADE ON DELETE SET NULL
);

CREATE TABLE EventProducts (
        ProductID INT AUTO_INCREMENT,
  Theme ENUM ('Business', 'Casual', 'Holiday', 'Family'),
  Size INT,
  PriceRange ENUM ('$', '$$', '$$$', '$$$$'),
        Description TEXT,
  PlannerName VARCHAR(255),
  CONSTRAINT pk_EventProducts_ProductID PRIMARY KEY (ProductID),
        CONSTRAINT fk_EventProducts_PlannerName
                FOREIGN KEY (PlannerName)
    REFERENCES Planners (PlannerName)
                ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE DIYEvents (
        EventID INT AUTO_INCREMENT,
  Theme ENUM ('Business', 'Casual', 'Holiday', 'Family'),
  Description TEXT,
  ListID INT,
  UserName VARCHAR(255),
  CONSTRAINT pk_DIYEvents_EventID PRIMARY KEY (EventID),
        CONSTRAINT fk_DIYEvents_ListID
                FOREIGN KEY (ListID)
    REFERENCES ElementLists (ListID)
    ON UPDATE CASCADE ON DELETE SET NULL,
        CONSTRAINT fk_DIYEvents_UserName
                FOREIGN KEY (UserName)
    REFERENCES DIYers (UserName)
    ON UPDATE CASCADE ON DELETE SET NULL
);
```

```
CREATE TABLE Reservations (
        ReservationID INT AUTO_INCREMENT,
  CreateTime TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  EventDate TIMESTAMP,
        UserName VARCHAR(255),
  PlannerName VARCHAR(255),
  CONSTRAINT pk_Reservations_ReservationID PRIMARY KEY (ReservationID),
  CONSTRAINT fk_Reservations_UserName
                FOREIGN KEY (UserName)
    REFERENCES DIYers (UserName)
    ON UPDATE CASCADE ON DELETE CASCADE,
        CONSTRAINT fk_Reservations_PlannerName
                FOREIGN KEY (PlannerName)
    REFERENCES Planners (PlannerName)
    ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE Reviews (
        ReviewID INT AUTO_INCREMENT,
  CreateTime TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  Rating FLOAT (2, 1),
  Content LONGTEXT,
  UserName VARCHAR(255),
  ProductID INT,
  CONSTRAINT pk_Reviews_ReviewID PRIMARY KEY (ReviewID),
  CONSTRAINT fk_Reviews_UserName
                FOREIGN KEY (UserName)
    REFERENCES DIYers (UserName)
    ON UPDATE CASCADE ON DELETE SET NULL,
        CONSTRAINT fk_Reviews_ProductID
                FOREIGN KEY (ProductID)
    REFERENCES EventProducts (ProductID)
    ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE Comments (
        CommentID INT AUTO_INCREMENT,
  CreateTime TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  Content LONGTEXT,
  UserName VARCHAR(255),
  EventID INT,
  CONSTRAINT pk_Comments_CommentID PRIMARY KEY (CommentID),
  CONSTRAINT fk_Comments_UserName
```

```
                    FOREIGN KEY (UserName)
        REFERENCES DIYers (UserName)
        ON UPDATE CASCADE ON DELETE SET NULL,
            CONSTRAINT fk_Comments_EventID
                    FOREIGN KEY (EventID)
        REFERENCES DIYEvents (EventID)
        ON UPDATE CASCADE ON DELETE CASCADE
);

#load values for tables
LOAD DATA INFILE '/Users/xinyuwang/Data/Restaurants.csv' INTO TABLE Restaurants
            FIELDS TERMINATED BY ',' ENCLOSED BY '""'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Wines.csv' INTO TABLE Wines
            FIELDS TERMINATED BY ',' ENCLOSED BY '""'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Movies.csv' INTO TABLE Movies
            FIELDS TERMINATED BY ',' ENCLOSED BY '""'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Musics.csv' INTO TABLE Musics
            FIELDS TERMINATED BY ',' ENCLOSED BY '""'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Gifts.csv' INTO TABLE Gifts
            FIELDS TERMINATED BY ',' ENCLOSED BY '""'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/ElementLists.csv' INTO TABLE ElementLists
            FIELDS TERMINATED BY ','
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Persons.csv' INTO TABLE Persons
            FIELDS TERMINATED BY ','
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES;
```

```
LOAD DATA INFILE '/Users/xinyuwang/Data/Administrators.csv' INTO TABLE Administrators
        FIELDS TERMINATED BY ','
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Users.csv' INTO TABLE Users
        FIELDS TERMINATED BY ','
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/DIYers.csv' INTO TABLE DIYers
        FIELDS TERMINATED BY ','
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Planners.csv' INTO TABLE Planners
        FIELDS TERMINATED BY ','
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/DIYEvents.csv' INTO TABLE DIYEvents
        FIELDS TERMINATED BY ','
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/EventProducts.csv' INTO TABLE EventProducts
        FIELDS TERMINATED BY ','
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES;

LOAD DATA INFILE '/Users/xinyuwang/Data/Reviews.csv' INTO TABLE Reviews
        FIELDS TERMINATED BY ',' ENCLOSED BY '"'
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES
  (ReviewID, Rating, Content, UserName, ProductID);

LOAD DATA INFILE '/Users/xinyuwang/Data/Reservations.csv' INTO TABLE Reservations
        FIELDS TERMINATED BY ','
  LINES TERMINATED BY '\n'
  IGNORE 1 LINES
  (ReservationID, EventDate, UserName, PlannerName);

LOAD DATA INFILE '/Users/xinyuwang/Data/Comments.csv' INTO TABLE Comments
```

```
        FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 LINES
(CommentID, Content, UserName, EventID);
```

Row count of each table

root@localhost
EventSmart.Administrators

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**2**
Table rows:	**10**

root@localhost
EventSmart.Comments

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**5**
Table rows:	**2807**

root@localhost
EventSmart.DIYers

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**1**
Table rows:	**4490**

root@localhost
EventSmart.DIYEvents

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**5**
Table rows:	**5001**

root@localhost
EventSmart.ElementLists

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**6**
Table rows:	**5001**

root@localhost
EventSmart.EventProducts

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**6**
Table rows:	**5000**

root@localhost
EventSmart.Gifts

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**5**
Table rows:	**9254**

root@localhost
EventSmart.Movies

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**5**
Table rows:	**5135**

root@localhost
EventSmart.Musics

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**4**
Table rows:	**73979**

root@localhost
EventSmart.Persons

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**3**
Table rows:	**4975**

root@localhost
EventSmart.Planners

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**2**
Table rows:	**500**

root@localhost
EventSmart.Reservations

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**5**
Table rows:	**1999**

root@localhost
EventSmart.Restaurants

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**6**
Table rows:	**12950**

root@localhost
EventSmart.Reviews

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**6**
Table rows:	**1731**

root@localhost
EventSmart.Users

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**4**
Table rows:	**5301**

root@localhost
EventSmart.Wines

Table Details

Engine:	**InnoDB**
Row format:	**Dynamic**
Column count:	**5**
Table rows:	**120365**