**Final Project**

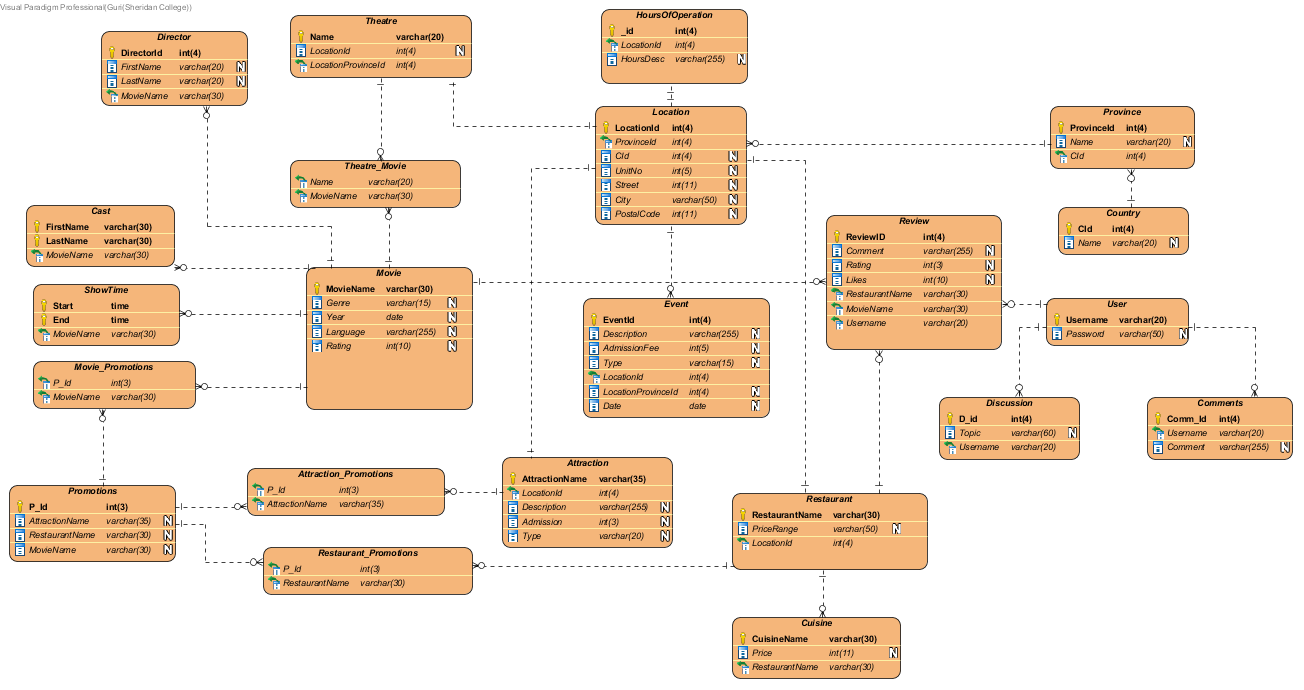
**Due: December 9th, 2018**

**Group Members: Inderpreet Singh Matharu and Gurcharan Singh**

**Instructor: Basil Burgess**

## **Entity Relationship Diagram**

## **Data Structure Diagram**



## **Normalization Report**

## **Movie Table:**

1NF:

+MovieName, Genre, Cast, Year, Director, Language, Rating, Showtime

2NF:

+MovieName, Genre, Language, Showtime

+Director, +MovieName, Rating

3NF:

+MovieName, Genre, Language, Rating

+DirectorID, FirstName, LastName

+MovieName, +Cast, FirstName, LastName

+Showtime, Start, End

## **Location Table:**

3NF:

+LocationId, ProvinceId, UnitNo, Street, City, PostalCode, Cid

## **Hours Of Operation Table:**

1NF:

+\_id, LocationId, +StartDate, +EndDate, Day, StartTime, EndTime

2NF:

+\_id, LocationId

+StartDate, +EndDate, Day

+StartDate, Day, StartTime, EndTime

+EndDate, Day, StartTime, EndTime

3NF:

+\_id,LocationId

+StartDate, +Day

+EndDate, +Day

+Day, StartTime, EndTime

## **Restaurant Table:**

1NF:

+RestuaurantName, Cuisine, +Price, LocationId

3NF:

+RestaurantName, LocationId

+Price, Cuisine

## **CREATE STATEMENTS FOR ALL TABLES**

Create database project;

Use project;

CREATE TABLE HoursOfOperation (

\_id int(4) NOT NULL AUTO\_INCREMENT,

LocationId int(4) NOT NULL,

HoursDesc varchar(255),

PRIMARY KEY (\_id));

CREATE TABLE Theatre (

Name varchar(20) NOT NULL,

LocationId int(4),

LocationProvinceId int(4) NOT NULL,

PRIMARY KEY (Name));

CREATE TABLE Movie (

MovieName varchar(30) NOT NULL,

Genre varchar(15),

Cast varchar(255),

Year date,

Director varchar(255),

Language varchar(255),

Rating int(10),

ShowTime time,

PRIMARY KEY (MovieName));

CREATE TABLE Province (

ProvinceId int(4) NOT NULL AUTO\_INCREMENT,

Name varchar(20),

CId int(4) NOT NULL,

PRIMARY KEY (ProvinceId));

CREATE TABLE Country (

CId int(4) NOT NULL AUTO\_INCREMENT,

Name varchar(20),

PRIMARY KEY (CId));

CREATE TABLE Event (

EventId int(4) NOT NULL AUTO\_INCREMENT,

Description varchar(255),

AdmissionFee int(5),

Type varchar(15),

LocationId int(4) NOT NULL,

LocationProvinceId int(4),

`Date` date,

PRIMARY KEY (EventId));

CREATE TABLE Review (

ReviewID int(4) NOT NULL AUTO\_INCREMENT,

Comment varchar(255),

Rating int(3),

Likes int(10),

RestaurantName varchar(30) NOT NULL,

MovieName varchar(30) NOT NULL,

Username varchar(20) NOT NULL,

PRIMARY KEY (ReviewID));

CREATE TABLE `User` (

Username varchar(20) NOT NULL,

Password varchar(50),

PRIMARY KEY (Username));

CREATE TABLE Discussion (

D\_id int(4) NOT NULL AUTO\_INCREMENT,

Topic varchar(60),

Username varchar(20) NOT NULL,

PRIMARY KEY (D\_id));

CREATE TABLE Restaurant (

RestaurantName varchar(30) NOT NULL,

PriceRange varchar(50),

LocationId int(4) NOT NULL,

PRIMARY KEY (RestaurantName));

CREATE TABLE Comments (

Comm\_Id int(4) NOT NULL AUTO\_INCREMENT,

Username varchar(20) NOT NULL,

Comment varchar(255),

PRIMARY KEY (Comm\_Id));

CREATE TABLE Attraction (

AttractionName varchar(35) NOT NULL,

LocationId int(4) NOT NULL,

Description varchar(255),

Admission int(3),

Type varchar(20),

PRIMARY KEY (AttractionName));

CREATE TABLE Promotions (

P\_Id int(3) NOT NULL AUTO\_INCREMENT,

AttractionName varchar(35),

RestaurantName varchar(30),

MovieName varchar(30),

PRIMARY KEY (P\_Id));

CREATE TABLE Attraction\_Promotions (

P\_Id int(3) NOT NULL,

AttractionName varchar(35) NOT NULL);

CREATE TABLE Theatre\_Movie (

Name varchar(20) NOT NULL,

MovieName varchar(30) NOT NULL);

CREATE TABLE Movie\_Promotions (

P\_Id int(3) NOT NULL,

MovieName varchar(30) NOT NULL);

CREATE TABLE Restaurant\_Promotions (

P\_Id int(3) NOT NULL,

RestaurantName varchar(30) NOT NULL);

CREATE TABLE Location (

LocationId int(4) NOT NULL AUTO\_INCREMENT,

ProvinceId int(4) NOT NULL,

CId int(4),

UnitNo int(5),

Street int(11),

City varchar(50),

PostalCode int(11),

PRIMARY KEY (LocationId));

CREATE TABLE Director (

DirectorId int(4) NOT NULL AUTO\_INCREMENT,

FirstName varchar(20),

LastName varchar(20),

MovieName varchar(30) NOT NULL,

PRIMARY KEY (DirectorId));

CREATE TABLE Cast (

FirstName varchar(30) NOT NULL,

LastName varchar(30) NOT NULL,

MovieName varchar(30) NOT NULL,

PRIMARY KEY (FirstName,

LastName));

CREATE TABLE ShowTime (

`Start` time NOT NULL,

`End` time NOT NULL,

MovieName varchar(30) NOT NULL,

PRIMARY KEY (`Start`, `End`));

CREATE TABLE Cuisine (

CuisineName varchar(30) NOT NULL,

Price int(11),

RestaurantName varchar(30) NOT NULL,

PRIMARY KEY (CuisineName));

ALTER TABLE Province ADD CONSTRAINT FKProvince778277 FOREIGN KEY (CId) REFERENCES Country (CId);

ALTER TABLE Discussion ADD CONSTRAINT FKDiscussion45867 FOREIGN KEY (Username) REFERENCES `User` (Username);

ALTER TABLE Comments ADD CONSTRAINT FKComments265086 FOREIGN KEY (Username) REFERENCES `User` (Username);

ALTER TABLE Review ADD CONSTRAINT FKReview649198 FOREIGN KEY (RestaurantName) REFERENCES Restaurant (RestaurantName);

ALTER TABLE Review ADD CONSTRAINT FKReview795509 FOREIGN KEY (MovieName) REFERENCES Movie (MovieName);

ALTER TABLE Review ADD CONSTRAINT FKReview994891 FOREIGN KEY (Username) REFERENCES `User` (Username);

ALTER TABLE Attraction\_Promotions ADD CONSTRAINT FKAttraction716818 FOREIGN KEY (AttractionName) REFERENCES Attraction (AttractionName);

ALTER TABLE Attraction\_Promotions ADD CONSTRAINT FKAttraction303676 FOREIGN KEY (P\_Id) REFERENCES Promotions (P\_Id);

ALTER TABLE Theatre\_Movie ADD CONSTRAINT FKTheatre\_Mo842225 FOREIGN KEY (Name) REFERENCES Theatre (Name);

ALTER TABLE Theatre\_Movie ADD CONSTRAINT FKTheatre\_Mo985785 FOREIGN KEY (MovieName) REFERENCES Movie (MovieName);

ALTER TABLE Movie\_Promotions ADD CONSTRAINT FKMovie\_Prom303790 FOREIGN KEY (P\_Id) REFERENCES Promotions (P\_Id);

ALTER TABLE Movie\_Promotions ADD CONSTRAINT FKMovie\_Prom941137 FOREIGN KEY (MovieName) REFERENCES Movie (MovieName);

ALTER TABLE Restaurant\_Promotions ADD CONSTRAINT FKRestaurant869816 FOREIGN KEY (RestaurantName) REFERENCES Restaurant (RestaurantName);

ALTER TABLE Restaurant\_Promotions ADD CONSTRAINT FKRestaurant360853 FOREIGN KEY (P\_Id) REFERENCES Promotions (P\_Id);

ALTER TABLE HoursOfOperation ADD CONSTRAINT FKHoursOfOpe293675 FOREIGN KEY (LocationId) REFERENCES Location (LocationId);

ALTER TABLE Theatre ADD CONSTRAINT FKTheatre406742 FOREIGN KEY (LocationProvinceId) REFERENCES Location (LocationId);

ALTER TABLE Attraction ADD CONSTRAINT FKAttraction651857 FOREIGN KEY (LocationId) REFERENCES Location (LocationId);

ALTER TABLE Event ADD CONSTRAINT FKEvent695728 FOREIGN KEY (LocationId) REFERENCES Location (LocationId);

ALTER TABLE Restaurant ADD CONSTRAINT FKRestaurant354477 FOREIGN KEY (LocationId) REFERENCES Location (LocationId);

ALTER TABLE Location ADD CONSTRAINT FKLocation436230 FOREIGN KEY (ProvinceId) REFERENCES Province (ProvinceId);

ALTER TABLE Cuisine ADD CONSTRAINT FKCuisine479277 FOREIGN KEY (RestaurantName) REFERENCES Restaurant (RestaurantName);

ALTER TABLE ShowTime ADD CONSTRAINT FKShowTime92528 FOREIGN KEY (MovieName) REFERENCES Movie (MovieName);

ALTER TABLE Cast ADD CONSTRAINT FKCast599533 FOREIGN KEY (MovieName) REFERENCES Movie (MovieName);

ALTER TABLE Director ADD CONSTRAINT FKDirector4460 FOREIGN KEY (MovieName) REFERENCES Movie (MovieName);

## **INSERT STATEMENTS FOR ALL TABLES**

## **Theatre Table:**

INSERT INTO `project`.`Theatre` (`Name`, `LocationId`, `LocationProvinceId`)

VALUES ('Silver City', '456', '786');

## **Promotions Table:**

INSERT INTO `project`.`Promotions` (`P\_Id`, `AttractionName`, `RestaurantName`, `MovieName`) VALUES ('989', 'Museum', 'St. Johns', 'Avengers');

## **Attraction Table:**

INSERT INTO `project`.`attraction` (`AttractionName`, `LocationId`, `Description`, `Admission`, `Type`) VALUES ('Gallery', '486', 'Art Gallery', '$20', 'Art');

## **Event Table:**

INSERT INTO `project`.`event` (`EventId`, `Description`, `AdmissionFee`, `Type`, `LocationId`, `LocationProvinceId`, `Date`)

VALUES ('551', 'Sport Event', '$10', 'Sport', '123', '554', '15/12/2018');

## **Review Table:**

INSERT INTO `project`.`review` (`ReviewID`, `Comment`, `Rating`, `Likes`, `RestaurantName`, `MovieMovieName`, `Username`)

VALUES ('345', 'Excellent', '99', '1989', 'St. Louis', 'Hulk', 'Porthos');

## **Province Table:**

INSERT INTO `project`.`province` (`ProvinceId`, `Name`, `CId`)

VALUES ('221', 'Ontario', '11');

## **Country Table:**

INSERT INTO `project`.`country` (`CId`, `Name`)

VALUES ('21', 'Canada');

## **User Table:**

INSERT INTO `project`.`user` (`Username`, `Password`)

VALUES ('Aramis', 'xyz@786');

## **Discussion Table:**

INSERT INTO `project`.`discussion` (`D\_id`, `Topic`, `Username`)

VALUES ('33', 'Weather', 'Athos');

## **Comments Table:**

INSERT INTO `project`.`comments` (`Comm\_Id`, `Username`, `Comment`)

VALUES ('41', 'Feron', 'Awesome');

## **Director Table:**

INSERT INTO `project3`.`director` (`DirectorId`, `FirstName`, `LastName`, `MovieName`)

VALUES ('675', 'John', 'Wang', 'SpiderMan');

## **Cast Table:**

INSERT INTO `project3`.`cast` (`FirstName`, `LastName`, `MovieName`)

VALUES ('Tony ', 'Stark', 'IronMan');

## **ShowTime Table:**

INSERT INTO `project3`.`showtime` (`Start`, `End`, `MovieName`)

VALUES ('7', '10', 'Justice League');

## **Cuisine Table:**

INSERT INTO `project3`.`cuisine` (`CuisineName`, `Price`, `RestaurantName`)

VALUES ('Indian Cuisine', '55', 'Montana Bar And Grill');

## **Movie Table:**

INSERT INTO `project3`.`movie` (`MovieName`, `Genre`, `Year`, `Language`, `Rating`)

VALUES ('Transformers', 'Science & Technology', '2015', 'English', '99');

## **Location Table:**

INSERT INTO `project3`.`location` (`LocationId`, `ProvinceId`, `CId`, `UnitNo`, `Street`, `City`, `PostalCode`)

VALUES ('21', '12', '11', '913', 'Edenbrook Hill', 'Brampton', 'L7A2X4');

## MongoDB Selection Reasoning

1. Discussion: There are numerous users and quite a bit of comments, if mySQL will be used it would be just to slow to access all the topics and comments, hence noSQL is used.
2. Movie: Movie entity has a lot of attributes, furthermore movies have reviews which further link to ratings and comments, hence to increase speed i.e. to have a better user experience noSQL is implemented.
3. Promotion: As movie, dinner and attraction is required for a single promotion and the join doesn’t have more than 1 single layer it would be easier and would have more speed than SQL.

The main reason to use mongoDB or noSQL is to increase speed i.e. have a better user experience.

# **Creating Collections:**

## **Discussion:**

db.createCollection(‘Discussion’);

## **Movie:**

db.createCollection(‘Movie’);

## **Promotion:**

db.createCollection(‘Promotion’);

# **Inserting Documents:**

1.

db.Discussion.insertOne(

{\_id:361

User:”John”

Topic:”Best Restaurant in GTA”

Comment: [{\_id:123, user:”Max”, comment:”St. Johns is the best.”}

{\_id:321, user:”Tony”, comment:”You can never go wrong with Boston Pizza.” }]

}

);

2.

db.Movie.insertOne(

{name:”Sahara”,

genre:”Comedy”,

Cast:[{\_id:12, name:”Matt”}

{\_id:44, name:”Angelina”}]

Year:2018,

Director:”John”,

Language:”French”,

ShowTime: [{13:20} {15:00} {20:10}]

Rating:”75%”,

Review:[

{\_id:111, user:”Kelly”, comment:”Nice”, rating:”85%”, likes:50}

{\_id:222, user:”Preet”, comment:”It was ok”, rating:”70%”, likes:35}

]

}

);

3.

db.Promotion.insertOne(

{

movieID:34, restaurantID:11, attractionID:89

}

);