PHASE 3 – IMPLEMENTATION

By: Lucas Gullo, Geoffrey Splendorio, Randi Tinney

COMP 4410 Database Management

Mahmood Hossain

November 29, 2018

**Purpose:**

We are to create a program for the company, *Movies-R-Us*. We are tasked to design a backend database for the company’s users and inventory. The users will be able to login to the front end Java application with their registered email address and their password. Once in they will be able to see all of the carried games and movies and, depending on the member level they paid for, will be able to rent a certain number of the available forms of entertainment. They will be able to search movies by actor, director, genre, sequels, and keywords. They will also be able to filter by award winning movies and movies that they haven’t checked out previously. They will be able to search games by genre, platform, and keywords. Finally, they will be able to see their rent history which will include all of the details of the movie/game, check out dates, and return dates.

There will also be specific users called administrators that will be able to add/remove members, movies, and games. They will also be able to update inventory details. They will be able to view the items that were rented in the last 24hours which will include the order details and shipping information. Finally, they will be able to see the top ten items based on number of rentals for the past month.

Within the database, the inventory will include movies and games. Each movie and game within the database will include the following details: title, cast, director, release date, genre, platform, version, awards won, and the immediate sequel if one exists. Information for all of the cast and director will include their name and address. Finally, each game and movie will include the total number of copies available and the current number of copies available.

Within the database, each user will include the following information: email, name, address, phone#, password, their quota for rentals, and past rental history.

**Assumptions:**

For this project, we assumed that an administrator would automatically be a member by design. This means that an administrator is a user with all of the same attributes.

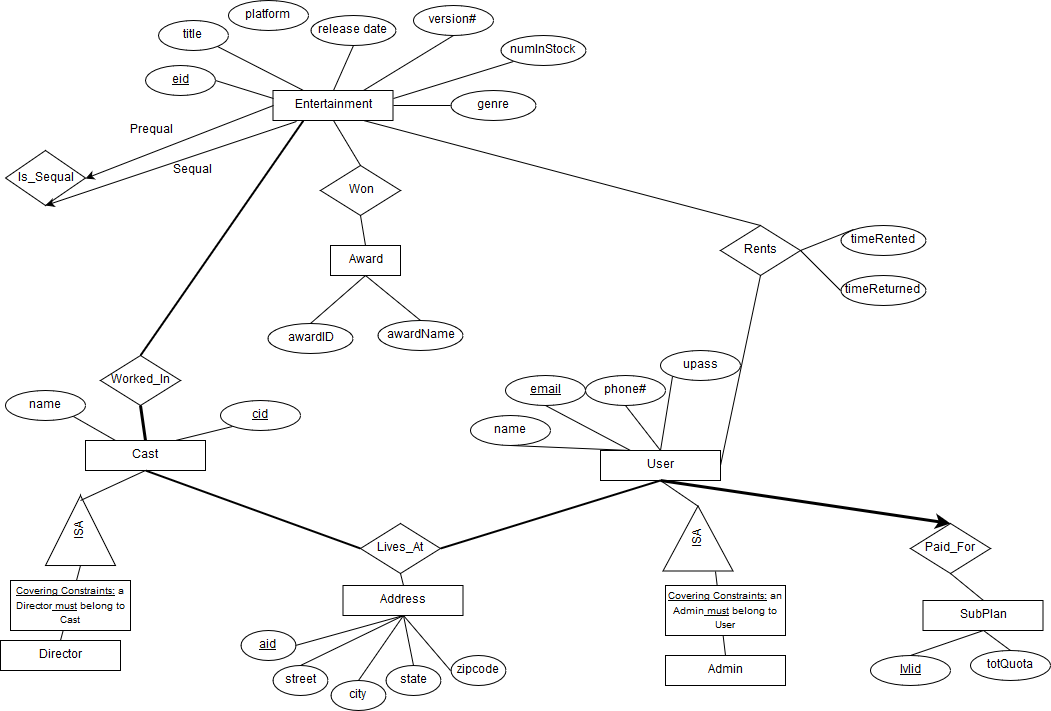
We assumed that each form of entertainment can only have one direct sequel.

For the keyword search, we are only going to be searching for keywords within the title.

Within the ER diagram, we do not have the various permissions represented. For example, we do not have represented that an administrator can add movies or games or that they can change their information. We do, however, have represented that users can rent the forms of entertainment.

For the top 10 forms of entertainment, we assumed that each month starts on the 1st at 12:00am. This means that when the clock rolls over on this day and the database is started up again, the previous month’s information will be wiped, the past month will take its place, and the number of rentals will be set to 0 to start recording for this month. We are also assuming that when we say “past month” we are truly meaning the past month and not the current month. For example, if it is October, we want to see the top 10 rentals for September, not what is top 10 currently for October. Then, when it changes to November, the information for September will be wiped, the information for October will take its place, and we will begin recording information for November.

**ER Diagram:**

****

**Relational Schema:**

**Entertainment (eid, title, release\_date, genre, num\_in\_stock, sequal\_id, platform, version)**

* “eid” is a Primary Key
* “title” is NOT NULL
* “genre” is NOT NULL
* “num\_in\_stock” is NOT NULL
* “platform” is NOT NULL
* “version” is NOT NULL
* “sequal\_id” Foreign Key to Entertainment

**Rents (rid, eid, uid, time\_rented, time\_returned)**

* “rid” is a Primary Key
* “eid” is a Foreign Key to Entertainment
* “uid” is a Foreign Key to User
* “time\_rented” is NOT NULL
* “time\_returned” is NOT NULL

**Users (email, name, pass, phone, aid, is\_admin, level\_id)**

* “email” is PRIMARY KEY
* “level\_id” is a FOREIGN KEY to Sub\_Plan Table (NOT NULL)
* “aid” is a FOREIGN KEY to Address Table (NOT NULL)
* “name” is NOT NULL
* “pass” is NOT NULL

**Sub\_Plan (level\_id, total\_quota)**

* “level\_id” is a Primary Key
* “total\_quota” is NOT NULL

**Cast (cid, name, aid, is\_director)**

* “cid” is a Primary Key
* “name” is NOT NULL
* “aid” is a Foreign Key to Address Table (NOT NULL)
* “is\_director” is NOT NULL

**Worked\_In (cid, eid)**

* “cid” and “eid” make a composite Primary Key
* “cid” is a Foreign Key to Cast (NOT NULL) (ON DELETE/UPDATE CASCADE)
* “eid” is a Foreign Key to Movies (NOT NULL) (ON DELETE/UPDATE CASCADE)

**Address (aid, state, city, street, zip)**

* “aid” is a Primary Key
* “state” is NOT NULL
* “city” is NOT NULL
* “street” is NOT NULL
* “zip” is NOT NULL

**Awards(awardID, title)**

* “awardID” is a Primary Key
* “title” is NOT NULL

**Won(awardID, eid)**

* “awardID, eid” is a Primary Key
* “awardID” is a Foreign Key to Awards
* “eid” is a Foreign Key to Entertainment

**Screenshots:**

**Queries:**

**User Search By:**

1. **Actor**
2. **Director**
3. **Genre**
4. **Keywords**
5. **Platform**
6. **Award Winning Movies**
7. **Award Winning Games**
8. **Movies the User Has Not Checked Out Yet**
9. **Games the User Has Not Checked Out Yet**

**User Views List of All Sequels**

**User Views Detailed Rent History**

**Admin View All Entertainment Rented in Last 24 Hours**

**Admin View Top Ten Entertainment Rentals of Last Month**

**Transactions:**

**User Rent Entertainment**

**User Modify Personal Information**

**Admin Add Member**

**Admin Remove Member**

**Admin Add Entertainment**

**Admin Remove Entertainment**

**Admin Update Inventory**

**System’s Limitations:**

Currently, there is no synopsis of the various movies or games within the system. This is often a useful thing to have when one is looking for a movie or game if they are not sure what the form of entertainment is about. Thus, if a user within our system finds a movie or game with an interesting title, they would have to go onto the internet to get a description.

Our system also does not give the language the forms of entertainment are in. Foreign films are rather popular among some groups of people and they might like to both search by and rent various entertainment by the language they are in or the country they are from.

Further, our system does not have a way to search by the company that created the entertainment. For instance, users cannot search for Disney movies or games made by Ubisoft. This seriously hampers some of our users in their searching and renting process.

Our system does not allow users to filter out the forms of entertainment that we do not have in stock at the time. If it did, it could quicken the search time for users and allow them to find their desired entertainment faster. Also, our system does not allow for users to store forms of entertainment into a queue if said entertainment is out of stock. Say, for instance, a user wants movie A, but A is out of stock. The user cannot put A into a wish list or queue that allows them to rent it as soon as A becomes available. The user is forced to constantly checking back until A is available and hope they hit the rent button faster than another user.

**Possible Improvements:**

Beyond including the limitation as discussed previously, another improvement includes adding a rating system to the movies and games. This would be based on the users’ own feelings towards the movie or game and they would be able to rate it based on either a star rating or percent rating. This would allow users to search by which movie or game is more popular among other users and base their rentals upon that if they so desire. Along this same line of thought, we could also add a type of review area where users can write their thoughts on the entertainment. This could also be useful for users when they are trying to make their rental decisions.

Another improvement could be an entertainment request area. If the system does not have a form of entertainment that the user wants, they can request to have it added. If enough users request the same entertainment, then it tells the admins that this entertainment is a highly requested item and we should add it to the stock.

Finally, another possible improvement could be an alert system of some kind to inform users of any new arrivals that are added to the system. We would also add a search feature that allows users to search by new arrivals so they can see what entertainment is new to the system.