CIS550-Project Milestone 2

Letian Chu, letianc@seas.upenn.edu
Weicong Dai, weicongd@seas.upenn.edu
Ziyang Luo, luoziy@seas.upenn.edu
Junyongzhao, junyong@seas.upenn.edu

March 27, 2020

1. Motivation for the idea and description of the problem the application solves

Do you know what is the most valuable thing in the tough time of corona virus? According the data from Amazon, the most needed thing in the pandemic time is the Switch rather than the toilet paper or the sanitizer. Our website provides the information about the game people play every day to help users have a better understand to the game they play and recommend the potential games they may be interested. Also, users can store the games they like by creating an account.

2. List of features you will definitely implement in the application

- ***Search***

In the search page, user input the name of the game, we return the information of the game including its developer, its release year and so on. every time the user inputs a game, such as the 2K20 or FIFA20, the website can display the game and the details of the game. Users can also filter the game search result based on the category of the game, rating or ESRB ratings.

- ***Recommendation***

The application makes recommendations based on the user's needs and hobbies. For instance, if the user likes the gun or the weapon, we recommend the GTA5 to him. And if the user likes the mythology, the application will recommend him the Witcher.

- ***Comparison***

The user could specify two games by user input and compare the information difference between two games, such as launch date, platform, price and reviews, etc. This will be a dynamic query which take user input in the front-end and provide information based on demand in the back-end.

3. List of features you might implement in the application, given enough time

- ***Store and Account***

Every user wants to have his/her own space to store the movie they are interested and put the movies into his/her own accounts. And the movies in the users' accounts can be used as the baseline in the Recommendation.

4. List of pages of the application

• Home/Search page

The home page will also be the search page, and the user can enter queries in a search bar. Information regarding the game will be returned on the same page below the search.

• Recommendation page

There will be the drop-down menus for users to select their goals (genre, name, developers, platform, etc.). Also, they can choose if the recommendation can use their searching history. Also, they can adjust their prefer like choosing not seeing some results anymore.

• Store band Account page (given time)

In this page, every user will see different data based on if they had add the game to his/her account. Also, they can choose their favorite games here to show their tastes.

5. Relational schema as an ER diagram

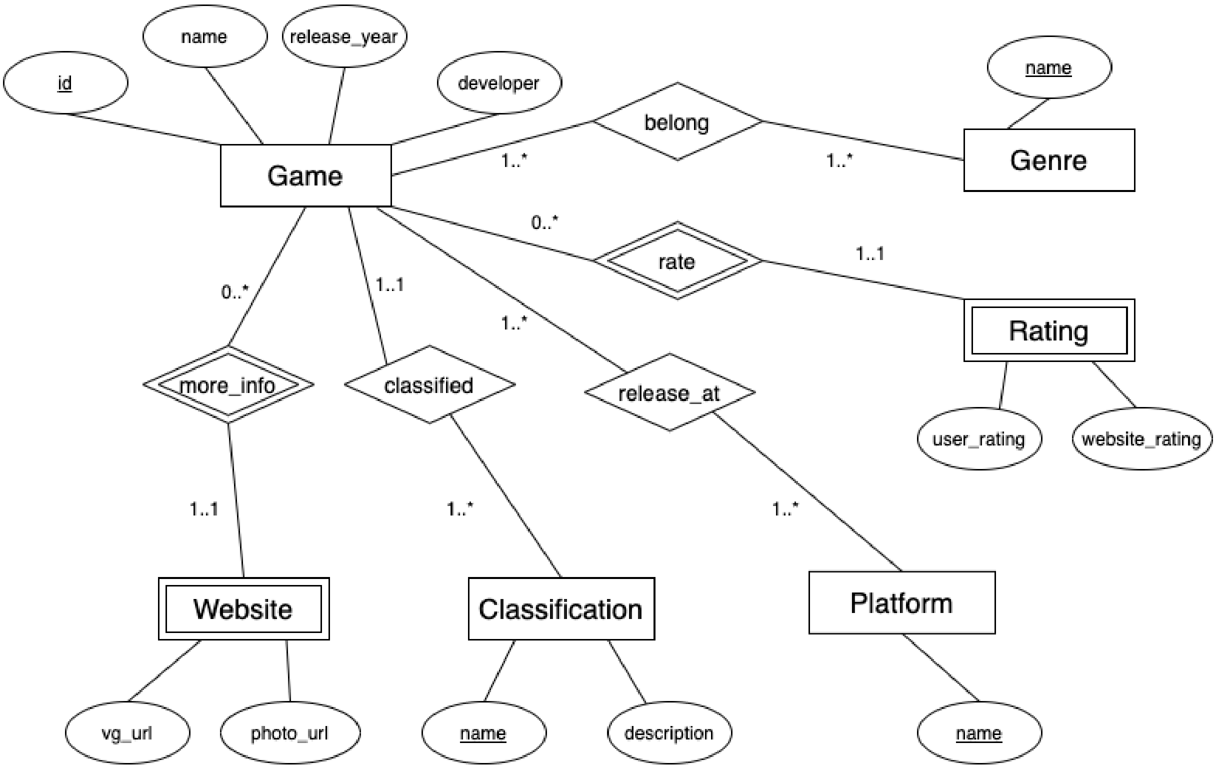

Figure1: ER diagram

6. SQL DDL for creating the database

```
CREATE TABLE Game (
   id INT(5),
   name VARCHAR(20) NOT NULL,
   release_year INT(4) NOT NULL,
   developer VARCHAR(20) NOT NULL,
   PRIMARY KEY (id)
)
CREATE TABLE Genre (
   name VARCHAR(20),
   PRIMARY KEY (name)
)
CREATE TABLE Game_genre (
   game_id INT(5),
   genre_name VARCHAR(20),
   PRIMARY KEY (id, genre_name),
   CONSTRAINT FOREIGN KEY game_id REFERENCES Game(id),
   CONSTRAINT FOREIGN KEY genre_name REFERENCES Genre(name)
)
CREATE TABLE Rating (
   id INT(5),
   user_rating DECIMAL(3),
   website_rating DECIMAL(3),
   PRIMARY KEY (id)
)
CREATE TABLE Platform (
   name VARCHAR(20),
   PRIMARY KEY (name)
)
CREATE TABLE Game_plat (
   game_id INT(5),
   plat_name VARCHAR(20),
   PRIMARY KEY (id, plat_name),
   CONSTRAINT FOREIGN KEY game_id REFERENCES Game(id),
   CONSTRAINT FOREIGN KEY plat_name REFERENCES Platform(name)
)
CREATE TABLE Classification (
   name VARCHAR(20),
   description VARCHAR(256),
   PRIMARY KEY (name)
)
CREATE TABLE Game_class (
   game_id INT(5),
   class_name VARCHAR(20),
   PRIMARY KEY (id, class_name),
   CONSTRAINT FOREIGN KEY game_id REFERENCES Game(id),
   CONSTRAINT FOREIGN KEY class_name REFERENCES Classification(name)
)
CREATE TABLE Website (
   id INT(5),
   vg_url VARCHAR(256),
   photo_url VARCHAR(256)
   PRIMARY KEY (id)
)
```

7. Explanation of how to clean, pre-process and ingest data into database

•.The download data are in 5 CSV files, each is a unique table with multiple attributes. We first pre-process the data by selecting the CSV file with no missing data to use, then filter out the attributes we will use in our application.

• We plan to clean the data by first defining a form for storing various types of attributes, such as using lowercase for string format and using float for digital. Then we will write specific python function to check the Integrity of the data belongs to those attributes.

•We also will check the consistency with the ER diagram we define for our attributes between the relationship of each entity.

•Then we will run command to ingest our data with the table we define into database.

8 List of Techniques in Project

- MySQL, MongoDB,
- Python, HTML, CSS, JavaScript
- Node.JS, AngularJS/React, AWS

Note: the techniques that will be used in this project is subject to the change to the any future needs.

9 Responsibility of each group member

- Weicong Dai: frontend and data cleaning

- Junyong Zhao: backend and connecting to cloud

- Ziyang Luo: frontend and connecting backend to frontend

- Letian Chu: backend and testing