Final Report

Nintendo Switch Game

Munira Hassan

Munira.hassan@go.winona.edu

Martin Liu

Mliu15@winona.edu

Department of Computer Science

Winona State University

CS 385

Dr. Zhang

November 29,2021

Introduction

Structure Query Language (SQL) is a database query language for storing and managing data in Relational Database Management Systems (RDBMS). For E.F Codd's relational database model, SQL was the first commercial language. SQL is the standard database query language in almost all RDBMS today (MySql, Oracle, Informix, Sybase, MS Access). In an RDBMS, SQL is utilized to perform all forms of data operations. SQL is used to perform create, retrieve, update, and delete operation on rational databases. It can perform administrative tasks on database such as database security, data backup manages huge amount of data, export/ import and create tables inside database. For our project we had to identify a problem and develop a working relational database system.

Our partner and I did Nintendo Switch Game because we had a common interest. We will get database from all the games that are well commonly known and getting customer reviews. We will be explained how we did it from scratch throughout the paper.

Proposal

Before we started to design our database system, we needed to write a proposal for our database system. We needed to write down problems that needed to be address, how questions are answered with the system, what attributes and constraints needed for our database. Some query question we have are what are SQL query to find the cost of each Nintendo Switch games? How many people bought the games? What are the release dates of the Nintendo switch games? What games were bought mostly? What vendor had the games? These question where what we were trying to solve. Some of the SQL constraints we thought we needed to use for our database is create table construct, insert, where, into, form, foreign key, primary key, join, natural join, in, and string operation. We ended up using only table construct, insert, into, from, foreign key, primary key, where not null and string operation for our database. The entities we brainstorm to use were customer, games, sale\_customer, sales\_vendor and vendor. The attributes we will use for customer will be customer\_name, date birth, emails. For customer\_ID (games, game\_id, title, year\_rel, price, review, rating) For sale\_customer (saleID, gameID, customerID, quantity, sale\_date, and sale\_price) For sales\_vendor (saleID, gameID, vendorID, quantity, sale\_date, and sale\_price) For vendor (vendor\_name, vendorID, email, and location). How we will obtain our data is having a select statement to retrieve records from the database.

The role each team member contributed were my partner and I agreed to split everything in half in this project. For milestone I, we divided the work in half. There were five question and we split that by Martin getting three questions and Munira getting two questions. For milestone II, we constructed five entities, we split that in three of two. Where Martin got customer, games, sale\_customer and Munira got sales\_vendor and vendor. For PowerPoint we split that in half. For the ER design, Martin did the design. For our final report, we split that in half and each person did two pages. Our partner and I made sure if we end up having any question, we will help each other.

Milestone II

After we brainstorm our proposal and database. We ended up designing an entity relationship diagram (ER). An Entity Relationship (ER) Diagram is a form of flowchart that shows how entities within a system, such as people, things, or concepts, interact with one another. In the disciplines of software engineering, business information systems, education, and research, ER Diagrams are most commonly used to build or troubleshoot relational databases. They employ a predetermined collection of symbols such as rectangles, diamonds, ovals, and connecting lines to illustrate the interconnectedness of entities, relationships, and their properties, and are also known as ERDs or ER Models. Entities are nouns, and relationships are verbs, mirroring linguistic structure. We had to create a relation database ER diagram that’s made up of two or more tables of information which are connected some way. We had five entities. So we created five tables and added the attributes n each table inside. We will discuss one table to give an broad explanation. For the figure 1, it shows all the entities. For game table, which included information about each attribute.

Diagram

Description automatically generated

Figure 1, ER Diagram

A picture containing text, whiteboard

Description automatically generated

Figure 2, Coronation

Data Set

Text, letter

Description automatically generated

Figure 3, Data Set

In figure 3, it shows our data set. You can see we insert our videodata into the database and generated into our own database instead of real database. As you can see, we added four customers, added their DOB, age and their emails. We also added four popular games, the release dates, review of the customer and rating out of five. We used the vendor name of the location of the game, the vendor of the store, their emails and which state is located.

Cases

We generate our own database instead the actual real database. We had three cases we used, one case is getting list of games and their prices. As showing in figure 3, it shows the games and how much the price goes for. The query statement is,

Select title, price from game.

Table

Description automatically generated

Figure 3

The second case is getting the location that has sales for Pokémon. As showing in figure 4, it shows the location that has the game Pokémon. The query statement is,

Select Location from Vendor,Sales\_Vendor where Vendor.vendorId =

Sales\_Vendor.VendorId and Sales\_Vendor.GameID = '2001';

Table

Description automatically generated

Figure 4

The last case is getting the list all customers whose order quantality was greater than 1. As showing in figure 5, we have the customer who bought how many of the games. The query statement is

Select CustomerId , Quantity

from Sales\_Customer

where  Quantity > 1;

Table

Description automatically generated

Figure 5

Conclusion

When creating a database, there's one thing you must do first: plan ahead. Before you start working on the computer, look at the information you want to keep and the ways you want to obtain it, whether it's a single table or a series of tables. That's because a badly organized database can stymie you further down the road when you want to extract your data in a useful format. When the professor gave us the term project. Our partner and I was confused how will we do a database from scratch when in class the professor gave us a database called University that was stored information already. In class we would use that to learn. It was really helpful having to do the milestone I (proposal) because it made us brainstorm deeply and gave us kind of a visualize what our project will be about. SQL is important in the professional world because it helps display database and ideas to businesspeople/customers or other team members. An easy way to show the larger idea of things. Our thoughts on this whole term project from each member, Martin; It was having a good communicating skill and having time management with your team member. Munira; Since I have been sick lately, I would say having a time management was key for both of us and communication. We both managed to work together and figure out common grounds to get this project done.