COMP 3380 Project

Part 3 – Project Report

Group 21

April 11, 2023

Dhairya Kachhia

Gurdit Singh

Krish Patel

A summary of the data

We decided to work on data of a fictional rock band music company called Chinook. It provides access to various audio and video tracks from various artists. The dataset for this company was obtained from a publicly available database site [https://relational.fit.cvut.cz/dataset/Financial] and the dataset can be accessed using these credentials on MySQL Workbench:

```
hostname: relational.fit.cvut.cz
port: 3306
username: guest
password: relational
```

1) Why was it chosen?

   Our group wanted to work on some real-world situation and then we found dataset of Chinook which met our desire. While we were working with this dataset for our project, we felt like we were actually creating database design and interface for a company and that feeling motivated us to create the model with utmost detail and interest.

2) What does if consist of?

   The database contains information about various tables such as employees who work in the company, where they help customers who are purchasing audio and video track of various artists. The database also includes tables that contain information about customer's invoices, artist's albums and media type, playlist, and genre of a track.

3) How large is it?

   File size: 92 KB

   Total number of records: 11 tables and 1300+ rows altogether.

Discussion of the data model

Why was it broken down into those tables?

- Our database consists of information about a music band named Chinook which has multiple tables namely: Person, Employee, Customer, Invoice, InvoiceLine, Playlist, PlaylistTrack, Track, MediaType, Genre, Album, and Artist. The data is separated into these tables to promote a uniform and categorized distribution of data based on these parameters.

Did you face any difficult choices when designing the model?

- We faced a problem when we were creating the ER model which was to relate the entity "Track" with its respective related entities. This is because there are a lot of other entities in our model that are related to track, so we solved this issue by designing our ER model in such a way that the "Track" was placed in a corner to ensure that its relationships with other entities are clearly visible.

  Another challenge was finding correct participation of between various entities, especially between Track and InvoiceLine entity. To solve this problem, we checked that not every track was purchased by some customer, so there is a partial participation between Track and InvoiceLine entity.

Did the data model cleanly fit into the relational database?

- Yes, the data model cleanly fits into the relational database because the way we have ordered all the tables and their relationships avoids any anomalies and problems that need to be taken care of.

Do you regret any decisions you made in your model? Did you change your model from part 1 when implementing it in part 2? What changes, and why?

- No, we do not regret any decisions that we made in our model. We did not change our model from part 1 when we were implementing it in part 2. We made sure that all the tables were covered in the queries, and all the constraints were properly defined in our database just as they are shown in the Relational and ER model.

Database model

Could the data be modelled in a different way, why or why not?

- Being a database with only 11 tables and few relationships, it is less likely to model this data in a different way, but this dataset can be model in a slightly different way by removing the specialization of person and by giving all of its attributes to employee and customer entity. By eliminating person specialization, employee and customer entity will look clustered.

Given the work completed, would you choose this model?

- After the completion of our work, we were happy with our model because it was simplified, self-explanatory, and easy to work with.

List of some interesting queries

1) This query retrieves the first name, last name, and email of customers who made a purchase of $1.5 or more in the month of January, but not in the month of February, and

before the 15th day of that month. Additionally, the query returns the invoice date and total price of the customer's purchase, ordered by the ascending amount spent. This query helps identify customers who made purchases during a specific time frame and at a specific price point. For instance, as a company, this information would be useful in understanding customer behavior at the beginning of a new year or month.

2) This query retrieves the names of employees, the names of customers that the employees helped, the media type of the tracks that the customers bought, the number of years of experience of the employee, and the total number of sales that the employee made. This information is useful in analyzing employee performance, understanding customer preferences, and identifying trends in media-type sales. Additionally, it can determine if more experienced employees are able to sell more to customers.

3) This query identifies the top 10 customers who spent the most money on music purchases, along with the total amount spent and the most popular genre they purchased. This query helps identify the most valuable customers and determine the most popular music genres among them. This information can be useful in developing targeted marketing strategies for high-value customers and in stocking popular music genres in the company's inventory.

4) This query is designed to identify inactive customers by retrieving customers who haven't made any purchases in the last 6 months, excluding those who have never purchased anything. It provides valuable information on the date and amount of the customer's last purchase, as well as the number of months that have passed since then. By ordering the results in decreasing order of the number of months since the customer's last purchase, the query helps to prioritize customers for targeted re-engagement strategies. This can help businesses improve customer retention and increase revenue.

5) This query fetches the first name, last name, track name, artist name, and album name of customers whose first names begin with "F" or "R" and who have bought a track that starts with "A" or "B". The output is arranged in ascending order by track name. This query is beneficial in determining customer preferences and identifying trends in tracking sales based on customer name patterns.

Importance of relational database

Does this dataset require a relational database? Would other database system be a better choice in modelling this data? Why or why not?

- Yes, we think Chinook dataset requires a relational database. The reason for that is relational database is best to organize datasets. Other datasets like NoSQL could also be used but they can’t match the efficiency of the relational databases. This is because in relational databases the data is arranged in a way that fits into tables and connects different entities like customers, employees, and sales. Relational databases use a structured approach, which makes it easy to implement very complex queries as we did in the front-end interface of this project. However, if we had used any other database instead of relational databases, then it would have been very complex and difficult.

Would the “interesting queries” you wrote be easier or harder to re-create if you were using an alternative database?

- The "interesting queries" provided would be easier to recreate using a relational database, as the queries involve querying across multiple tables and joining them together based on their relationships. Relational databases are designed to handle such queries efficiently, making them a good fit for this type of analysis. Other types of databases may struggle with these types of queries, particularly if the relationships between entities are complex.

Interface Summary

Programming Language:

- The programming language used for the back end of this website is C#. Also, HTML and CSS were used to construct the front-end interface and add some designs.

Dependencies:

This website depends on the following dependencies:

- .NET Core Runtime: This is required to run the ASP.NET Core Web Application on the server. You can download it from the official .NET website.
- Microsoft.AspNetCore.App: This is the primary package that contains the ASP.NET Core framework and is required to run the website. It is automatically included when creating a new ASP.NET Core Web Application project in Visual Studio.

Libraries:

The following libraries were used in this website:

- Bootstrap: Bootstrap is a popular library/framework used in this project to create this website’s UI.

Summary:

- MusicBandInterface is a website that was built using ASP.NET Core Web Application with C# as the programming language. It depends on the .NET Core Runtime, and Microsoft.AspNetCore.App. It also uses Bootstrap to create a responsive and dynamic UI. The main purpose of this website is to show the Chinook database based on the query selected. There are 5 queries in the database, which can be selected using the blue button in the center of the screen.