

Andrea Zetter

May 9, 2018

CSC 411 Database

Bo Li

University of Southern Mississippi

**Abstract**

*Introduction*

The goal of this project was to provide us, the students, a realistic experience in the conceptual and logical design, implementation, operation, and maintenance of a relational database and associated applications. We were given many choices as to what our Database would represent, and I chose mine to mirror a Retailer’s Database. I chose this because I felt it would be widely applicable to the real world, even if a given business isn’t exactly a retailer business.

*Problem*

The only real problem presented to us is how to make this retailer database work. We have no previous knowledge of creating a database, but we have been given the necessary concepts and techniques from our course lectures, so all we need to do is apply them.

*Approach*

Before I even touched a computer, I made sure to think of what a retailer database would need to function properly and as expected. After that, I began to create ER diagrams, Relational Schemas, mock data, tests/queries, and optimization. I created a couple sketches of the ER diagram and as a team we went back forth as to what tables should be taken out and what is necessary to create our queries.

*Results*

Given that a project of this nature would take significantly longer than one college semester, I found my results to be acceptable by my standards, as well as my professors. Of course, my results were not correct from the start. It took some time and research to have created my database. Doing a project like this was difficult, but an effective way to learn the foundation for database creation, implementation, and maintenance. As a team we were also able to contribute to each other’s efforts, which it helped in many ways. Many times, we would get together and go over the issues we were having individually, and we would help each other improve our results and efficiently move forward.

*Conclusion*

From this, I have learned how to create, implement, and maintain a simple database. Could some aspects of this project have been implemented in a better way? Of course. The more time given for a project, the better. Due to the time frame it really made it difficult to create an effective database, but a lot was learned throughout the process. Working as a team really helped me see other frameworks and other ways of implementing this project, the three of us took a different route when we created our framework which was a learning experience for me.

*Problem Description*

To start things off, I knew I needed to map out what functionality a retailer database would need to have. For this project, we didn’t have to be too realistic with our functionality, but rather have a basic concept and be able to implement it in MySQL. In our project write-up, we were given this basic concept that everyone would know what the customer would want. With these requirements given to us, I began to think about how a retail company functions, or rather an online store would function. This occurred because I believed that we had to include these other concepts for our database to be correct in the eyes of our professor. I also had in mind that I had to create a full web application that simulates just like an online retailer. Having that thought process I figured I had to have an up and running web application that led customers to place orders as well creating an account as a customer.

While the professor did give us a starting point, I believed each function had some implied functionality to it, which we the students were to implement in our database structure. This problem was a self-created issue, after consulting the professor, it was made known that for the scope of this project we did not have to go as in depth as I anticipated or planned for. I didn’t just remove all the planned-out functionality from my project, but rather made a copy of it for later, so that on my own time I can work on it. Then I made adjustment to the project diagrams to reflect how I thought the basic functions of an online store should work. The next real issue was making sure I was using the proper MySQL statements to get the desired results. All that was needed to resolve this was time spent reviewing notes and information provided in the courses lectures.

*Database Design*

E-R Model:



*Relational Schema:*

Enterprise (ID, enterprise name, **inventory\_ID**, **store\_ID**)

Store(ID, storeName, storeAddress, storeState, storeZip, storeOpenTime, storeCloseTime, )

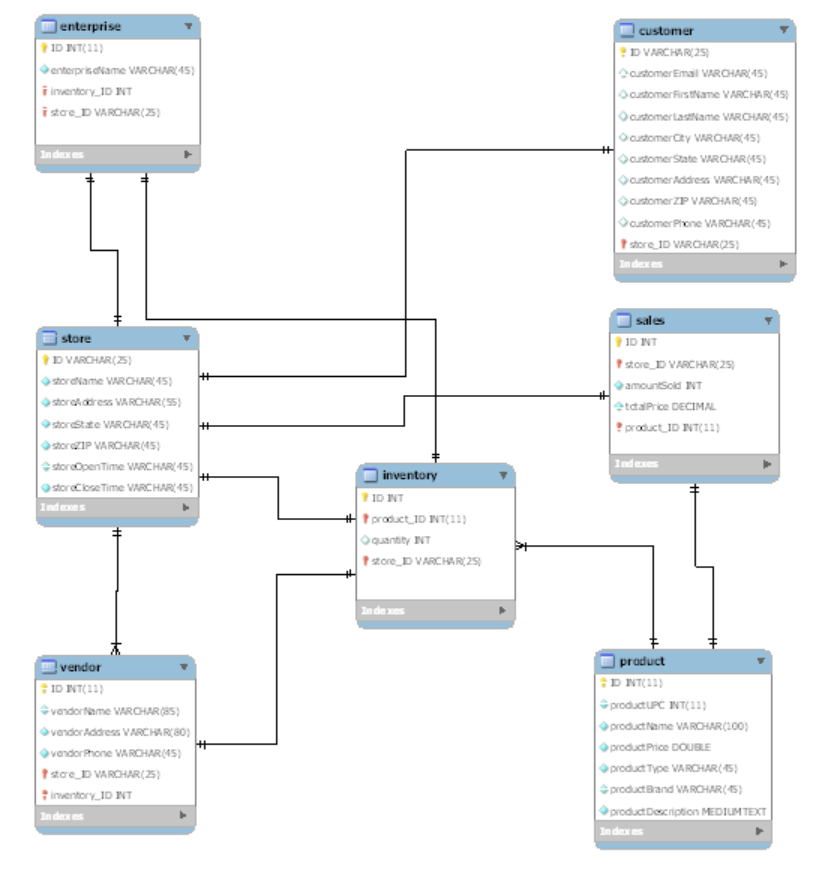
Vendor(ID, vendorName, vendorAddress, vendorPhone, **store\_ID, inventory\_ID**)

Inventory(ID, quantity, **store\_ID, product\_ID**)

Customer(ID, customerEmail, customerFirstName, customerLastName, customerCity, customerState, customerAddress, customerZip, customerPhone, **store\_ID**)

Sales(ID, amountSold, totalPrice, **store\_ID, product\_ID**)

Product(ID, productUPC, productName, productDescription, productPrice, productBrand, productType)



*Implementation Details*

Enterprise table was chosen since there is one enterprise that is responsible for many stores, and all its different stores refer to this enterprise. ID is the primary to identify the unique enterprise but can have more than one. Enterprise name is necessary to distinguish our different enterprises we created as a team.

Store table is needed to represent the many stores that are open within the same city or state. ID is used to identify each store. We included the store name, address, state, city, and zip to identify the store’s location as well as the time the store’s open and close time.

Vendor table is used to supply the store with merchandise. Vendor table has a relationship with store and inventory since it supplies the store and it should have access to view the store’s inventory stock. To better access what the store needs from a vendor.

Inventory table is connected to the store table to store a store’s inventory. Enterprise table is connected to inventory this way the enterprise can view any store inventory. This is also connected to product’s table, so inventory can identify what product’s each store carries. As an attribute we added quantity to know exactly how much of an item does a store carry. This will help with the queries.

Product table was created to identify different types of products that each store’s carry. Since not all store’s carry the same products. This table is connected to the inventory table to identify how much and what specific product is carried at a store.

Sales table was created to keep track of each store’s sales. Total price will help with how much total sales of each customer is being made and amount sold helps to know how much of a product was sold.

Customer table helps track what store customer shopped and store is connected to customer to help identify this. ID is the primary key to help distinguish each customer. Customer email, first name, last name, city, state, address, zip, and phone number helps the store distinguish its customers.

For the creation of the relational schema I used mySql workbench. HTML, CSS, and PHP were used to create my website. I also used XAMPP to connect the database to the website. I used draw.io to create the E-R model.

*Conclusion*

This project has taught me a lot about how complex a database can be. I’ve also learned about creating, implementing, and maintaining a MySQL Database. This would include the creating of an E-R diagram, relational schema, mock data, and a website, although the website was no a requirement. When approaching the E-R Diagram, I thought it was easy at first. However, I found that the way I thought something should work wasn’t always the way to represent it in a diagram which resulted in some changes to the diagram.

In conclusion, my experience with E-R Diagrams I would say that it is very theoretical. Entity relationships appear obvious at first but representing them properly became a bit more difficult. Relational schema was a bit easier, since it was created after the E-R diagram had been completed. Gathering the mock data was something new to me but was tedious more than anything else since a lot of it had to be edited and I created many problems for myself trying to edit the schema since mySql workbench does not like that at all. What made this tedious was that we were asked to keep a certain degree of realism in our data so that it would be easier for query analysis and more like a real-world task rather than a college project. The creation of a website was interesting and fun, since it was my first time doing so. There are many different approaches to making a website. As well as many different to connect database to the website. Connecting the database was tedious as well since it was my first time doing so and I had to play around until I got it to work. It also took a couple of how-to videos to help me understand the process.

*Read Me:*

* Created Relational Schema in MySQL workbench as well as tested my queries there as well.
* Used XAMPP to connect my database to the web application.
* Created one HTML web application where I have created links to direct to the queries
* Created 5 PHP pages to show queries.

*Screenshots*

