Term Project Iteration 6 (Jewels Can Tell)

Dhiraj Totwani

CS669

09/10/18

**Jewels Can Tell Overview**

I would like to create an inventory management application used on a Desktop for a small jewelry business that wants to track, manage, and assess its inventory. The name of this application would be named “Jewels Can Tell”. The jewelry store manager or owner can use the application to find, remove, or track any specific pieces of inventory. The application also provides the owner/manager with information about when the jewelry was acquired, groups the different types of jewelry based on materials, and track the popularity of specific pieces of jewelry. This application would empower a small jewelry business to not only make the right decisions in future purchases, but also a system to track and manage all the jewelry based on attributes related to them.

Here are some examples to how someone would use the Inventory Management Application. Garry who owns a small jewelry business had a spike in sales over the summer season. The application recorded the inventory that was sold including its attributes. Garry wanted to find out exactly what kind of jewelry sold the most. Garry opened Jewels Can Tell to find out what groups of items were sold the most. The engagement rings being at the top of the list made Garry aware of the most popular products in the summer. That information influenced Garry’s decision to purchase more engagement rings for the summer season.

There are many programming components to Jewels Can Tell, but for this course I will be focusing on the database components. The database will store the jewelry store’s inventory, and the inventory’s data (Material, Gemstones, price, categories, date acquired, date sold, quantity).

**Jewels Can Tell Use Cases**

**Purchase New Inventory Use Case**

1. Jewelry Store needs to purchase new inventory.
2. It buys jewelry from multiple Vendor/Suppliers.
3. The jewelry needs to be identifiable by the specific vender incase of damage or defect.

*Business Rule*

A supplier may provide multiple pieces of jewelry. A piece of jewelry can only be from one supplier.

**Customer Purchase Use Case**

1. Customer buys a few pieces of jewelry.
2. The purchase is stored in the system.
3. The customer comes back to upgrade a piece of jewelry.
4. The system searches up the initial purchase to give the exact credit value to add to the next purchase.

*Business Rule*

A customer can have 1 to many purchases. A sale can only be associated to one customer.

**Inventory Tracking Use Case**

1. A admin wants to view the quantity of engagement rings sold in the past month

2. The admin selects the option for engagement rings.

3. Jewels Can Tell displays the full list of engagement rings in and out of stock.

4. The admin clicks on the filters that were “amount sold”, and “last 30 days”.

5. Jewels Can Tell displays the amount of Engagement rings sold in the past 30 days.

6. The admin now can make an accurate analysis on the desirability of engagement rings.

*Business Rule*

Jewelry contains many different types. A piece of jewelry can only be of one type.

One to many employees can be assigned admins. A employee may or may not be an admin.

**Specific Inventory Tracking Use Case**

1. A customer walks in and asks if we have any rings with 2 carats of diamonds.
2. The admin opens Jewels Can Tell to search for the specific 2 Carat Rings.
3. The admin searches the jewelry type by ring in the database.
4. The database brings up all the rings with options to filter based on carat weight.
5. The admin then adjusts the carat weight filter to display only 1-2 carat weight diamond rings.
6. The admin then looks at the lowest and highest priced ring to determine a price range for the customer.
7. The customer now knows how much money is needed to purchase a 1-2 carat diamond ring.

*Business Rule*

A ring may or may not be diamond. A diamond may or may not be associated with a ring.

**Tracking Jewelry Sales and Quantity**

1. Jewelry store wants to track sales and quantity on hand.
2. Newly added items, and sold items need to be updated in the inventory.

*Business Rule*

A piece of jewelry exists within one inventory. An inventory contains one to many pieces of jewelry.

A piece of jewelry can be purchased many times. A purchase has one to many pieces of jewelry.

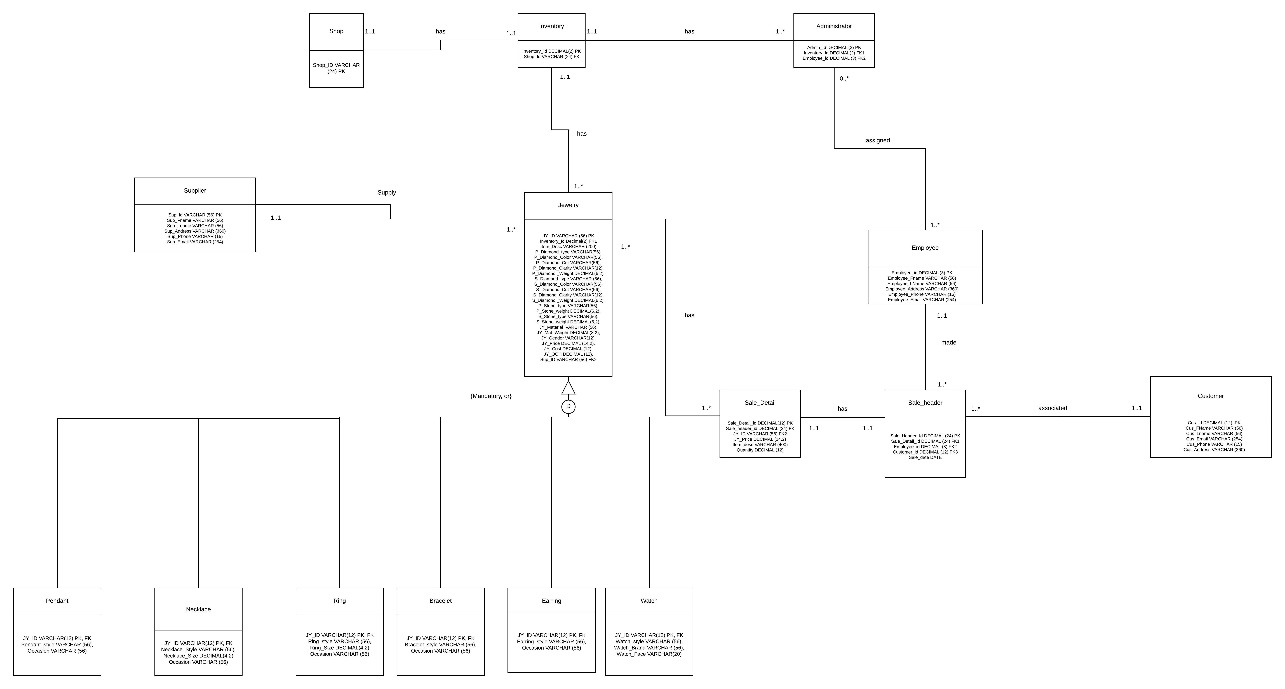
***All Structural Business Rules***

1. A piece of jewelry exists within one inventory. An inventory contains one to many pieces of jewelry. **Inventory/Jewelry 1:M Relationship**
2. A piece of jewelry can be associated to many sale orders. A sale order is only associated to 1 sale. **Jewelry/Sale Order** **1:M Relationship Sale Order/Sale** **1:1 Relationship**
3. A piece of jewelry is a necklace, ring, bracelet, earring, watch, or a pendant.
4. One to many employees can be assigned admins. An employee may or may not be an admin. (I will add an “Assigned” bridging entity if need be to solve this many to many relationship.)
5. A supplier may provide multiple pieces of jewelry. A piece of jewelry can only be from one supplier. **Supplier/Jewelry 1:M Relationship**
6. A customer can have 1 to many purchases. A purchase can only be associated to one customer. **Customer/Purchase 1:M Relationship**
7. The inventory can have one to many admins. An admin has access to only one inventory.

**Inventory/Admin 1:M Relationship**

1. One store can only have one inventory. An inventory can only be owned by one store.

**Store/Inventory 1:1 Relationship**



I will attach the high quality photo of this **UPDATED ERD** to the upload for clarity purposes.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Jewels Can Tell Data Tables**

Refer to the Updated Excel File with the Upload. The tables were to wide to fit into this document.

**Jewels Can Tell Index Identification**

|  |
| --- |
| **Primary Keys** |
| Store.Store\_id |
| Inventory.Inventory\_id |
| Jewelry.JY\_id |
| Ring.Jy\_id |
| Watch.Jy\_id |
| Earring.Jy\_id |
| Bracelet.Jy\_id |
| Necklace.Jy\_id |
| Bracelet.Jy\_id |
| Sale\_detail.Sale\_detail\_id |
| Sale\_header.Sale\_header\_id |
| Customer.cus\_id |
| Employee.Employee\_id |
| Administrator.admin\_id |
| Supplier.Sup\_id |

|  |  |  |
| --- | --- | --- |
| **Foreign Keys** | **Unique?** | **Description** |
| Inventory.Store\_id | Unique | There is only one store in the inventory. |
| Jewelry.Inventory\_id | Not | There can be many pieces of jewelry in the same inventory. Especially because there is only one inventory. |
| Jewelry.Sup\_id | Not | The same supplier can be supply items many times. |
| Ring.Jy\_id  Watch.Jy\_id  Earring.Jy\_id  Bracelet.Jy\_id  Necklace.Jy\_id  Bracelet.Jy\_id | Unique | The product is unique to itself and can only have 1 key. |
| Sale\_detail.sale\_header\_id | Unique | The sale detail can only have a Sale\_header\_id once. |
| Sale\_detail.jy\_id | Not | The piece of jewelry in the sale\_detail can be included on multiple sale\_details. |
| Sale\_header.Cus\_id | Not | The customer can be on many Sale\_headers |
| Sale\_header.Sale\_detail\_id | Unique | The sale header can only reference one sale detail\_id |
| Sale\_Header.Employee\_id | Unique | A sale\_header instance is unique to the employee. |
| Administrator.Employee\_id | Not | Many employees can be administrators. |
| Administrator.Inventory\_id | Not | Many administrators have access to the inventory |

I have added the Index script into the “JewelsCanTell” script in the end.

**PRICE CHANGE TRIGGERS**

I created a Price Change table and updated into my current ERD. That would be beneficial for me to change my prices back and forth during Christmas and holiday sales. That would also allow me to keep a log of alterations with the prices and when. I can also make sure that the administrators change the pricing in a timely manner.

I have included the screenshots into the folder as Price Change Triggers. The SQL for that is also included in my “JewelsCanTell” SQL in the last part.

