Zara Database

**Group 1**

Raquel Buezo Tordesillas A20410771

Eduardo Ballesteros Grande A20410861

Carlota López Pérez-Arroyo A20410752

Table of Contents

[Student’s Contribution 3](#_Toc498457375)

[Entity-Relationship Model 4](#_Toc498457376)

[Explanation and assumptions 4](#_Toc498457377)

[ER Diagram 5](#_Toc498457378)

[Relational Database 6](#_Toc498457379)

[Assumptions 6](#_Toc498457380)

[Relational Database Diagram 7](#_Toc498457381)

[SQL SCRIPT 8](#_Toc498457382)

[Application 9](#_Toc498457383)

# Student’s Contribution

# Entity-Relationship Model

## Explanation and assumptions

For our database project, we have chosen the widely known Spanish store called Zara. This store mostly concentrates on selling clothes, shoes and accessories. To create the Entity-Relationship Model, we have taken into account the following considerations:

* In the entity **Product:**
  + Size, color and item will be derived attributes form the *UPC\_code*
  + We consider for example that two shirts of the same size and color have the same *UPC\_code*
  + *Department* will be used to identify whether it belongs to the Man or Woman section.
  + *Type* will consider the subcategories into which the categories are divided. For example, we will consider dividing clothes into dresses, sweaters, pants, etc.
  + *Description* will be a brief explanation of the characteristics of the product
* In the entity **Store:**
  + *Opens and closes* refers to the schedule of the store
  + *Max\_inventory* is the maximum number of products that can be stored
* In the entity **Address:**
  + It is used to have a record of both the address of the customers and the stores
* In the entity **Category:**
  + We consider that each product would be separated into clothes, shoes and accessories
* In the entity **Vendor:**
  + *Delivery\_method* refers to the transportation system by which the product would be distributed. This includes train, truck, boat and airplane
  + *Delivery\_time* refers to the lead time (time interval since the store poses an order until it receives it)
  + We have considered that we have only one vendor per product
* In the entity **Customer:**
  + We consider that one customer can have several credit cards, so it is a multivalued attribute
* In the relationship **Transactions:**
  + We use it to keep a record of what each customer buys at each store and when it was bought.
* In the relationship **Inventory:**
  + *Price* is placed as an attribute in this relationship because it depends on the store.
  + *Threshold* is the quantity of each product, that when reached informs that a new order should be placed.
  + *Order\_quantity* is the quantity that the store orders of each product
* In the relationship **Employee:**

\*NOTE: The acronym CP found on the ER model, stands for “*Clave Primaria”* which is Spanish for Primary Key.

## ER Diagram

INSERT DIAGRAM HERE, I’M DOING THE EXPLANATION FOR THE LAST ONE

# Relational Database

## Assumptions

* We have used the data type “nvarchar” to be able to store UNICODE and multilingual data.
* The attribute UPC\_Code is a string where the first 9 characters identify the item, the next 2 are used to obtain the size and finally the remaining 3 represent the color.
* Delivery time in days and method truck/train/boat…
* Shoes no halves
* Unique size 00
* Store\_id=1 refers to internet which has max\_inventory=0 and opens at 00:00:00 and closes at 23:59:59
* Customer\_id=1 refers to anonymous but it must still enter an email and a password
* Customer may not give the address, so address\_id in Customer schema can take null
* If 2 people are living in the same house, they have different address\_id (?¿)

## Relational Database Diagram

THE LAST DIAGRAM

## SQL SCRIPT

In store, we have assigned store\_id=1 and max\_inventory=0 for the internet section

In Customer, we have assigned customer\_id=1 for the anonymous customer.

# Application

**LOCKS!! -> CHAPTER 9/10**

We’ve used C# in order to create a visual application that would let us interact with the database.

In our demo, the first thing that you have to do is select the store that you are at. In the real world, this process would not be necessary as the employee or the customer would have direct access to the Main form. We need the store location to obtain the store\_id. The reason behind this is that each store has different products.

[IMAGE OF THE MAIN FORM]

This would be the Main window of the application. In here you have different options:

* **Catalog:** lets the customers browse at the products that are in store and their price. If the product that they want is not in store it suggests other store locations where they may find it.

**[IMAGE]**

* **Log in:** it lets the customers log into their account. Once they’ve successfully logged in, they have two options. They can either modify their credit card numbers or they can modify their profile.

**[IMAGE, IMAGE, IMAGE]**

* **Register:** New customers can create an account. They would have to fill in ….. these are the compulsory fields.

**[IMAGE]**

* **Checkout:**
* **Employee:**