Creation of an Online Super Market Application

Project Report

Taleb Housseiny, Souheil Moussa

Abstract—This report details the creation and implementation of an online market. This market is tied to an online website where customers can place their orders and a desktop app where staff

CONTENTS:

1 INTRODUCTION:

This document is the project report of the Super Market project. This project consists in developing a DB and related software applications to store and manipulate the data of a Super Market. It demonstrates the database definition, manipulation, and the connection between the MySQL database management system and the applications. It also includes explanations regarding the desktop application that is done using Python and the website that is done using JavaScript. The supermarket has a DB containing details regarding the suppliers, products, units, shipments as well as employees. The DB stores the data that is provided by the supermarket admin or inventory employees. Inventory employees usually make shipments from the suppliers to get new products that are added automatically as units in the database. After making at least one shipment, customers by registering and logging to the website can order product units if available from the supermarket database.

2 BACKGROUND:

2.1 Context and Problem Analysis

The analysis of the objective in hand was done by first deciding what functionalities will the DB system be able to provide to different types of users (including admin, hr, and inventory).
The ***admin*** is
1) access all the information in the application
2) can add staff and make shipments
The ***inventory*** can:
1) Search for products (based on their name).
2) make shipments from suppliers
3) Check list of units.
The ***hr*** can:
1)Add, View and Delete Employees from the database
2) Add role for every employee

2.2 Information Needs

The information needs include information regarding books, related to mainly the title of the book, the author/s, year written, number of pages, and genre. Some information was also needed regarding the users (customers and libra-rains) including their names, emails, unique username, date of birth, etc.

3 PROPOSAL:

3.1 Conceptual Data Model (CDM):

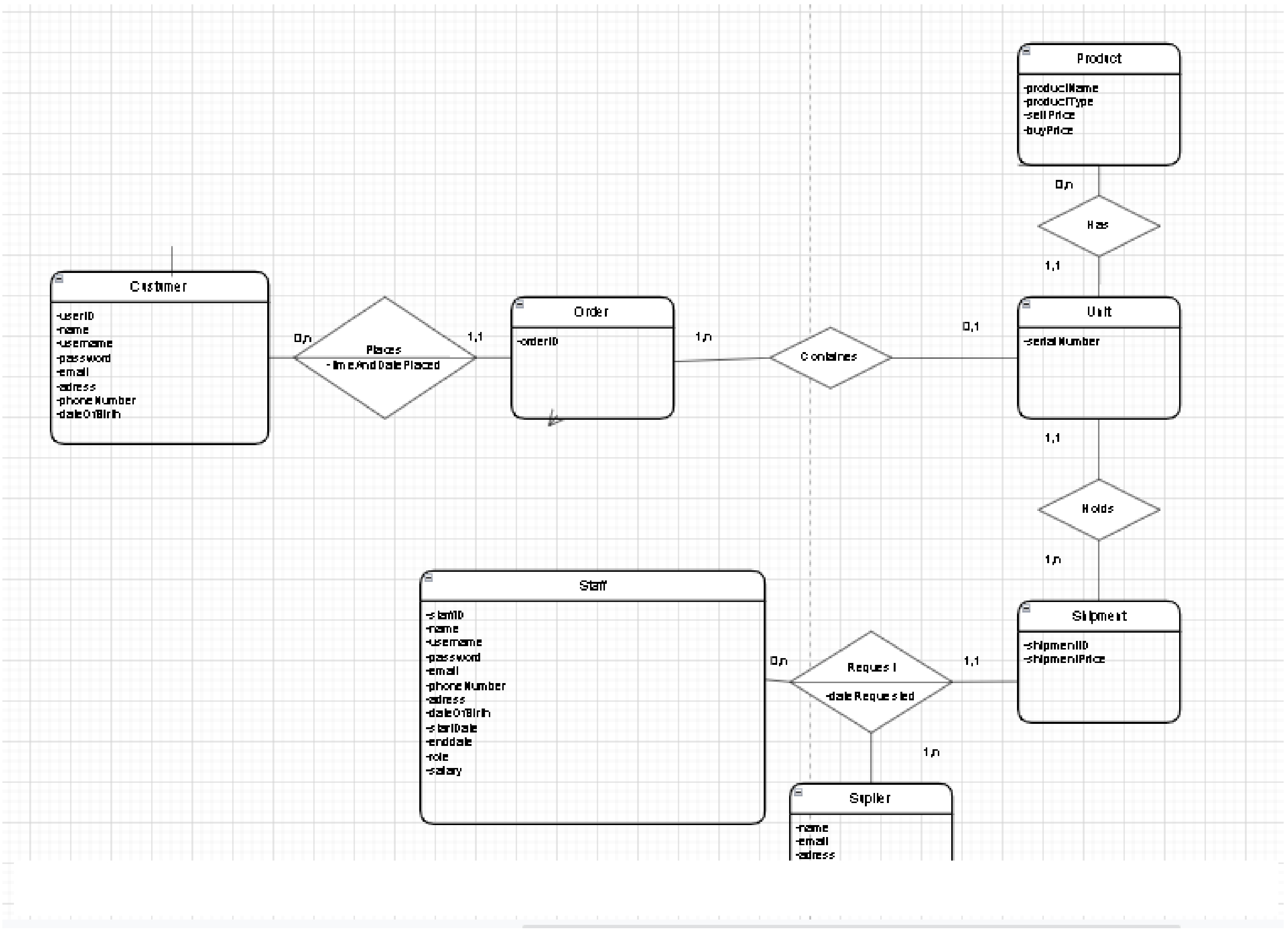

Figure 1Conceptual Data Model

Notes regarding the conceptual data model: 1.
Staff entity type: Admin, HR, Inventory manager have different roles. The username and password is used to login each Employee in the company.
2. Order entity type: the primary key is the order_id for order relation and it is custard for customer relation. Customers can order units and add them to their cart.
3. Shipments entity type: Admin, Inventory manger can make shipments from supplier by using staff id as a foreign key.

3.2 Logical Data Model (LDM):

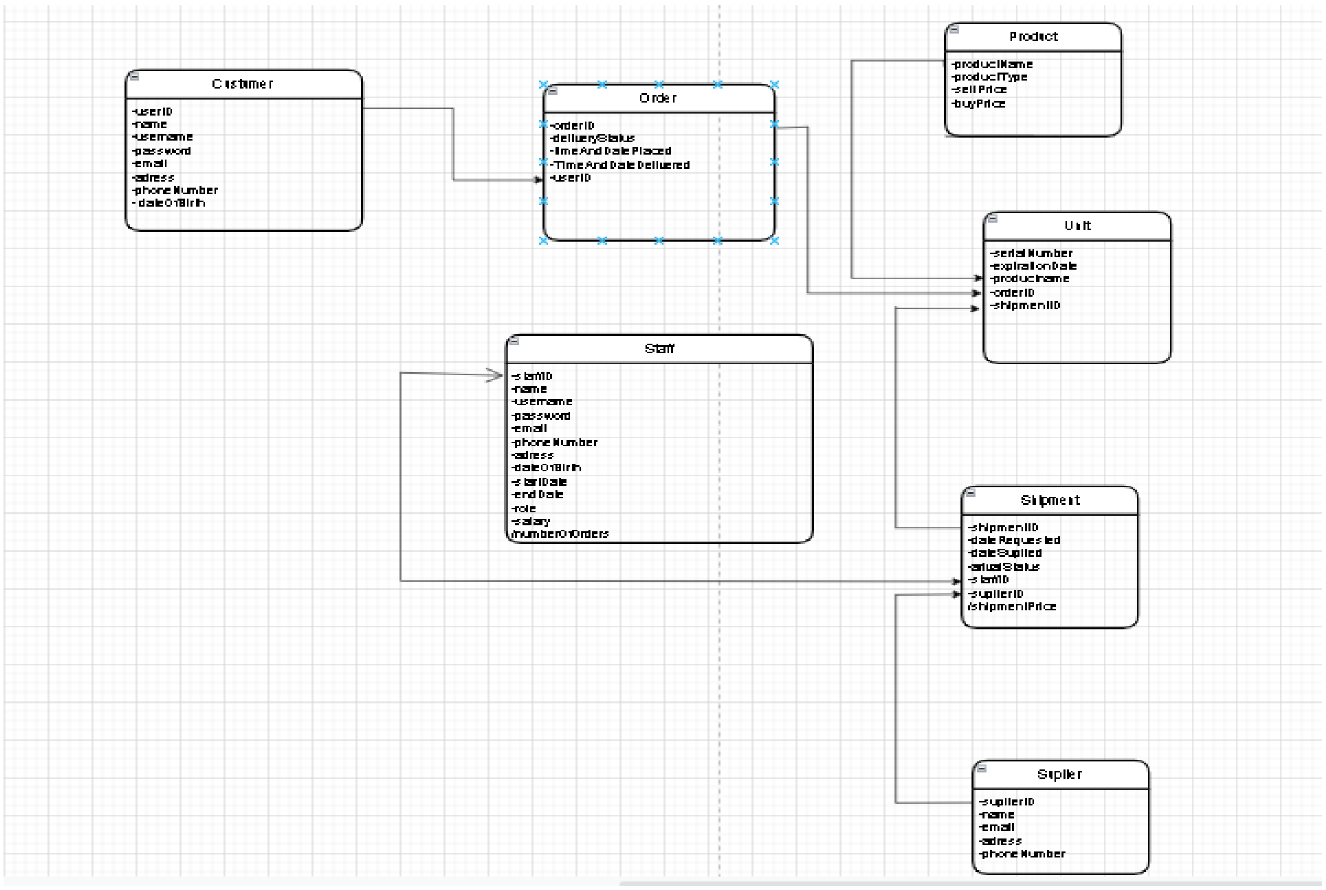

Figure 2Logical Data Model

The logical data model was obtained by simply converting the conceptual data model to a logical data model that was acquired during the class lectures. Order id is a foreign key between customer and order table and staff id is also a foreign key between staff and shipment relation. A staff member can make a shipment for a product from a supplier and identifying product unit quantity to ship.

3.3 Software Application and Database Design and Implementation:

The tools that were used in this project are the following:

1- MySQL Workbench
2- Express.js Framework
3- EJS
4- Tkinter

4 EXPEREMENTAL EVALUATIONS:

This section will provide query tests as both expert and non-expert users. Non-expert users are bounded to the ware interface where they provide a bunch of keywords, through textboxes or combo boxes. Expert users use SQL statements without the need of many text boxes to fill. This section will show the non-expert vs expert uses of the main functionalities of the desktop application.

a- Searching for a product, supplier name etc.:

Non-Expert: The non-expert user will be provided with a set of textboxes where he/she can specify the product name, supplier name and click the search button. The result will be a table showing all products satisfying the specified conditions. For example, assume the customer typed in ―Tuna ‖ as product name.

Expert: The above input will be translated into the following SQL statement: SELECT * FROM product WHERE product_name LIKE "%Tuna%.

b- Making a shipment:
Non-Expert: The non -expert user will be provided with a set of textboxes and combo boxes where user can enter all shipment data and then store them in the database.
Expert: The above functionality can be done by getting the product name, supplier name and quantity from the user and using SQL statements all other information will be initialized automatically and added to the database in both table shipments and units.
The SQL statement in python will be:

```
 sql1  =  "INSERT  INTO  `shipment`  (date_ordered,  staff_id,  ship-
ment_price, shipment_suplier) VALUES(%s, %s, %s, %s)"
    # (date_ordered, 'intel', 5*(select buy_price from product where
product_name= 'test1'),1)
    val3 = (date_ordered, rep1, price, sup,)
    cursor.execute(sql1, val3)
```

```
for x in range(qty):
        print(x)
                sql2   =   "INSERT   INTO   `unit`   (production_date,
unit_product_name, unit_shippment) VALUES(%s, %s, %s)"
        val4 = (date_ordered, pro, recent_shipment_id,)
        cursor.execute(sql2, val4)
```

c- Displaying Shipments:
Non-Expert: get all attribute values from the user and show them in a table (tree view).
Expert: Joining two tables in one SQL statement and showing the result in a tree view to get staff name in shipment table using staff_id as foreign key. The python code will be:

```
 cursor.execute("SELECT shipment_id, date_ordered, username, ship-
ment_price, shipment_suplier FROM shipment INNER JOIN staff ON ship-
ment.staff_id = staff.staff_id")
    fetch = cursor.fetchall()
    for data in fetch:
        tree.insert('', 'end', values=(data))
    cursor.close()
```

5 Conclusion:

This was the report of the Database System Course final project having the title “Super Market”. The different functionalities along with possible improvements were described in the report. As a personal experience, it is very important to have such projects since it develops my research skills which led us to acquire new skills when dealing with Express framework, python GUI and the connectivity between UI and the database.

REFERENCES

[1] FreeCodeCamp, "Python tkinter basics," 22 May 2019 [Online]
https://www.youtube.com/watch?v=YXPyB4XeYLA
[2] W3schools, "Bootstrap," [Online]
https://www.w3schools.com/bootstrap/
[3]W3schools," Python MySQL," [Online]
https://www.w3schools.com/python/python_mysql_getstarted.asp
[4]W3schools, "JavaScript," [Online]
https://www.w3schools.com/js/