

22018575

Database Systems Coursework Assignment – Idea Furniture Company

22018575

School of Computing and Mathematics, Keele University

CSC-20002 Database Systems

Dr Thomas Neligwa

May 19, 2024

Table Of contents

0 Introduction	3
1.0 Functional Requirements	3
1.1 Product Management:	3
1.2 Customer Management:	3
1.3 Order Management:	3
1.4 Store and Employee Management:	3
1.5 Supplier Management:	3
2.0 Database Design:	4
2.1 Products Table:	4
2.2 Customers Table:	4
2.3 Orders Table:	4
2.4 Stores Table:	5
2.5 Employees Table:	5
2.6 Suppliers Table:	5
2.7 Supplier Shipments:	5
3.0 Other Design Choices	
3.1 Quantity.....	7
3.2 Delivery Details & Collection Details	7
4.0 Database Implementation.....	7
4.1 No Redundant Purchase History Table:	8
5.0 Web application and Intergration	8
5.1 PHP Session Management	8
5.2 JavaScript Integration	9
5.3 Navigation Bar.....	9
5.4 Mascot	9
6.0 Conclusion	9

0 Introduction

The IDEA Company needs a database to manage its furnishing products, which vary significantly in size and weight as well as manage the customers' orders who are buying said products. This report outlines the functional requirements of the company, details the database design, implementation and web integration.

1.0 Functional Requirements

The database system for IDEA must address the following functional requirements:

1.1 Product Management

Store details of all products, including price, description, dimensions (for delivery items) and weight (for collection items).

Differentiate between products available for delivery and those for in-store collection.

1.2 Customer Management

Maintain unique customer accounts with personal details and purchase history.

1.3 Order Management

Record details of products purchased online, specifying delivery or collection requirements.

For delivery items: Manage and store details of delivery arrangements.

For collection items: They can be collected in store without prior arrangements.

1.4 Store and Employee Management

Maintain information on various stores across the country which includes store-specific employees.

Distinguish between employees responsible for delivery and those handling in-store collections.

1.5 Supplier Management

Keep records of suppliers associated with each store, including product deliveries from suppliers.

2.0 Database Design

The database design follows the relational model. Below is a summary of the key tables their relationships and ERD Diagrams:

2.1 Products Table

Fields: ProductID, Name, Description, Price, Quantity, Dimensions (nullable) and Weight (nullable).

Purpose: Store details of each product and Weight or dimensions are stored in their own tables with ProductID being both the primary and foreign key.

2.2 Customers Table

Fields: CustomerID, Name, Password, Email, Address and |Telno.

Purpose: Store customer personal information and unique identification.

2.3 Orders Table

Fields: OrderID, CustomerID, ProductID, StoreID, EmployeeID, OrderDate, quantity, DeliveryDetails (nullable) and Status.

Purpose: Record each purchase made by customers, specify the store responsible for the order, the employee processing the order, quantity ordered, status of the order and connect to the child table DeliveryDetails if the order is for delivery.

This table provides tracking No and courier to the customer.

2.4 Stores Table

Fields: StoreID, Location and name.

Purpose: Maintain information on each store's location.

2.5 Employees Table

Fields: EmployeeID, StoreID, Name, Role (Delivery/Collection) and Telno.

Purpose: Store details of employees, their roles, and the stores they are associated with.

2.6 Suppliers Table

Fields: SupplierID, StoreID, Name, Email and Telno.

Purpose: Record details of suppliers delivering products to each store.

2.7 Supplier Shipments

Fields: ShipmentID, SupplierID, ProductID , DateOfDeliverey and Quantity.

Purpose: Track shipments made by suppliers, including the products delivered and their quantities.

22018575

Below are the Conceptual and Logical ERD diagrams

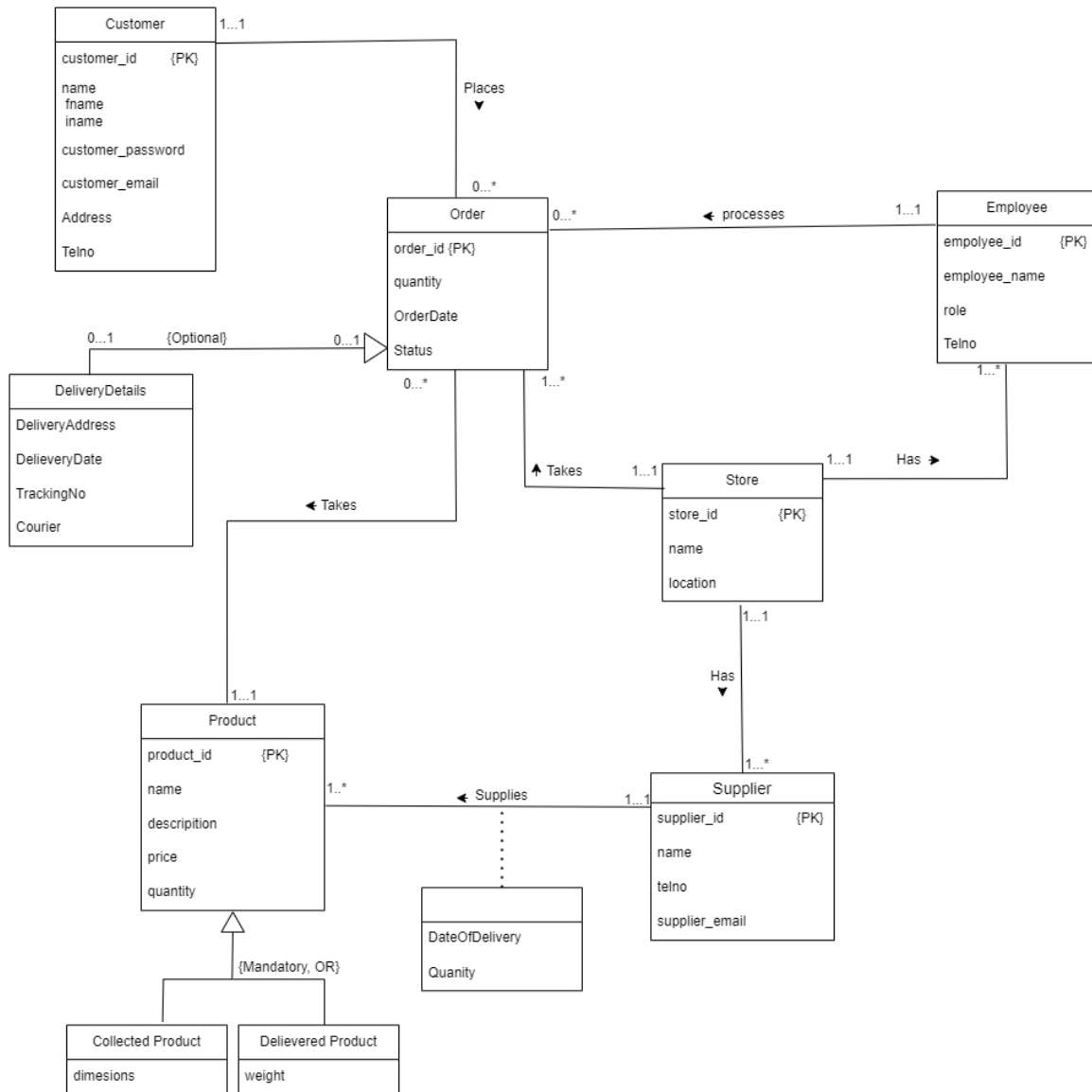


Figure 1 Conceptual ERD diagram

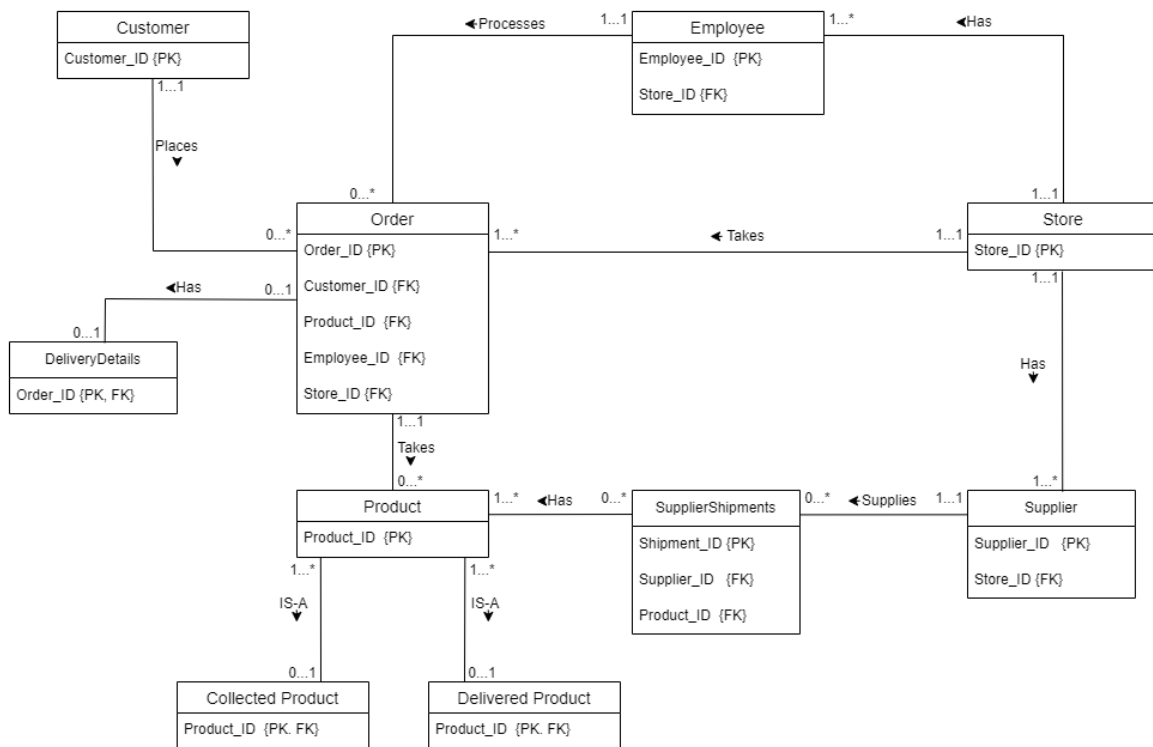


Figure 2 Logical ERD Diagram

3.0 Other Design Choices

3.1 Quantity

The quantity attribute allows the company to track inventory at both the store and company levels. This permits features such as locating the nearest store with available stock for customer collection.

3.2 Delivery Details & Collection Details

DeliveryDetails is a child table to Orders, allowing users to specify a delivery address and track the delivery status. Collection details are not needed as customers can collect items in-store with their order ID (Digital Receipt) during open hours.

4.0 Database Implementation

The implementation of this database is carried out using Oracle and SQL for database creation and management.

ON DELETE CASCADE was used for tables where the primary key is also a foreign key, such as `DeliveryDetails`, `CollectedProducts`, and `DeliveredProducts`. This ensures that related records are automatically deleted when a parent record is removed.

4.1 No Redundant Purchase History Table

A customer's purchase history is tracked using their `CustomerID` as a foreign key in the `Orders` table, eliminating the need for a separate purchase history table. SQL statements can be found in the occupying SQL file with filler data populated using ChatGPT.

5.0 Web application and Intergration

The IDEA website abstracts SQL commands to enhance usability for average users. While it supports dynamic searches, it does not include aggregate functions like **ORDER BY** or **GROUP BY**. Users can select any table from a dropdown menu, choose columns by clicking on their names (displayed as buttons) and perform partial match searches using **WHERE %\$search%**. For example, searching for customers named 'John' will display all matching results.

A design choice involves using **LEFT JOIN** to extend the details of `Collected Products` and `Delivered Products` with their respective product information when displaying their tables. This also expands delivery details with the order table. Although, it results in redundancy with repeating columns for the foreign key in the join.

5.1 PHP Session Management

The code works by starting a session in PHP and storing the chosen data in the session's ID, such as the current table name, search query and chosen column. There are **if** statements that check the current chosen column; if the column is selected, the query is added to the SQL statement using **sql .= "query"** for more flexibility.

5.2 JavaScript Integration

JavaScript is used to manage the dropdown box, allowing flexibility in table selection. If a new table is added to the database, it can be added to the `` through an `` with the value of the button in the `` needing to be replaced with the table's name. However, selection is based on the current **\$table** value and is printed out by PHP. This could pose a security risk for SQL injection attacks as exact table names are printed out.

5.3 Navigation Bar

The navigation bar is non-functional and is solely to demonstrate layout.

5.4 Mascot

IDEA's mascot, Meowchie, can be found in the top right corner, sporting a lightbulb in its chest representing IDEA. This enhances the UI and brand identity.

6.0 Conclusion

The designed database system for IDEA Company effectively meets the functional requirements, ensuring efficient product, customer, order, store, employee and supplier management. The implementation using SQL, PHP and JavaScript provides a flexible access. Future improvement would be to make specific views such as Customer and Staff where customers can view their orders and Employees can view the orders that they need to process.

Website

teach.scam.keele.ac.uk/prin/x7x92

Dropdown will throw an error upon first load however will work upon selecting a table [To-Fix]