

PRODUCT DESCRIPTION

Database Design & Implementation Experimental Case

Date:	05 December 2021
Product Title:	GetIt : An online store
Authors:	Jaskaran Singh (100399880) Prabjot Kaur (100397341) Sukhdeep Kaur (100399088) Japjot Singh (100396704)
Client:	INFO 2312 – KPU
Document Ref:	Product description
Version No:	0.5

1. PRODUCT DESCRIPTION HISTORY

1.1. Revision History

Revision date	Previous revision date	Summary of Changes	Changes marked
12/7/2021	12/04/2021	Modifications in the ER Diagram	
12/08/2021	12/07/2021	Change ER Diagram totally because unsatisfied with results	
12/08/2021	12/08/2021	Modified ER diagram more precisely and add Scripts and other definitions	

1.2. Distribution

This document has been distributed to:

Name	Title	Date of Issue	Version
Cesar Lopez Castellanos	Instructor	12/09/2021	0.5

Table of Contents

page

1. PRODUCT DESCRIPTION HISTORY	2
1.1. Revision History	2
1.2. Distribution	2
2. OVERVIEW	4
2.1. Identifier.....	4
2.2. Title.....	4
2.3. Purpose	4
2.4. Composition	5
3. USING THE PRODUCT	15
3.1. Special considerations	15
3.2. Instructions for use	15

2. OVERVIEW

2.1. Identifier

INFO2312 - Section A10 – Group -04

2.2. Title

GetIt – Online Store

2.3. Purpose

For ease to the client, the Database will display the product details, Customer can look through an item and its details just by putting the name of the product in the search bar which is provided for the easily access. It will show the number of products left in stock and whether the product is available or not for the user and when it is accessible to client. This database planned in a such manner that in case numerous clients can utilize it at the same time, it won't influence the running session of the site. Distinguishing the different clients' requirements and satisfying them in the proper manner.

This Data is accessed to everyone like different kinds of clients who are looking forward for their different choices such as regular customer, local shops. Apart from this, Businessman additionally team up with the suppliers and utilize the database.

Concerning the product's complexity, it is an easy-to-use, user-friendly solution that is not overly complex, allowing database operators and operator to readily obtain any available data and alter, insert, or delete it. Furthermore, because the product is built with the most recent software versions, it is quite resilient in terms of speed, security, and the possibility of encountering additional minor mistakes while running the programme.

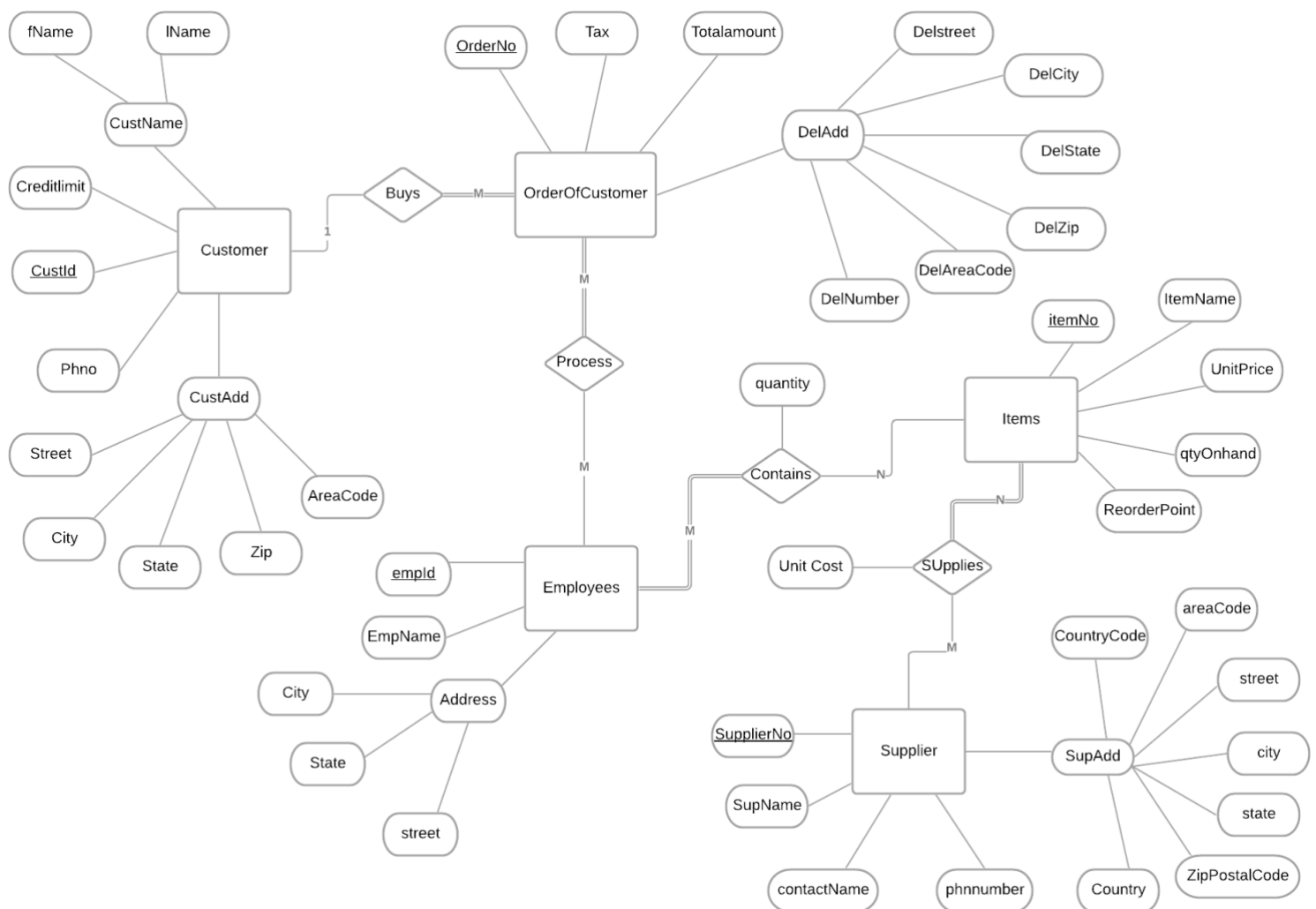
People who are familiar with the SQL language and the MySQL server software should be able to test and approve the product. They also require testing abilities and queries to determine whether they can add, alter, delete, and display existing data utilising SELECT, INSERT, UPDATE, OR DELETE commands. The testing process is straightforward, and the testing department must determine whether all four commands function on every table for every attribute.

The relational database management system (DBMS) was employed in this project to build the system. A relational database is a type of database that stores data in a structured manner using rows and columns. This database facilitates the search for and access to specific values inside the database. It is referred to as "relational" because the values in each table are related to one another. Tables can also be linked to each other. Because of the relational nature, queries can be conducted on several tables at the same time. For the long maintenance and storage of the data, Databased methodologies are used. The

database facilitates the separate process which assists in backup and restore the original data.

2.4. Composition

ER Diagram:



The ER Diagram is classified into 5 entities such as Item, Customer, OrderofCustomer, items, Supplier and Employees. The Item entity is categorized into itemid (Which is assigned as Primary key), and the attribute key are: ItemName, unit Price and other entities whereas in Customer table CustomerId is a primary key and detail of person from name to its address. As we come to supplier, it sells their item on the Online store and that table contain information about the supplier address from where they ship their items and supplierNo is primary key to identify the supplier uniquely. Employees are worker which are working for the online store and table elaborates which employee working for the specific table.

Contain is relationship entity which specify the orderNo, itemNo which are primary for the table where quantity tell how many items are inside of that order while supplies is also relationship entity which tell how many items are supplied by particular supplier and supplierNo, itemNo are primary keys.

Relational Schema:

Customer(custId, firstName, lastName, street, city, state, zip, creditLimit, areaCode, phnumber)

Orderofcustomer (orderNo, totalAmount, tax, DelStreet, DelCity, DelState, DelZip, DelAreaCode, DelNumber, custId)

Supplier(supplierNo, supName, countryCode, areaCode , phnumber, street, city, state, zipPostalCode, Country, contactName)

item(itemNo, itemName, unitPrice, qtyOnHand, reorderPoint)

employees(empld, EmpName, street, city, State, orderNo)

Contains(orderNo, ItemNo, Quantity)

supplies(SupplierNo, itemNo, UnitPrice)

Table 1. Definition of Table: Customer

Attribute	Datatype	Primary Key		Foreign Key		Referential Integrity				Additional Comments
		Yes	No	Yes	No	Parent		On Update	On Delete	
						Table	Attribute			
custId	Char	Yes			NO					
firstName	VarChar		NO		NO		custName			
lastName	VarChar		NO		NO		custName			
street	VarChar		NO		NO		custAdd			
city	VarChar		NO		NO		custAdd			
State	VarChar		NO		NO		custAdd			
Zip	Char		NO		NO		custAdd			
areaCode	Char		NO		NO		custAdd			
Credit limit	int		NO		NO					
phnumber	char		NO		NO					

Table 2. Definition of Table: OrderofCustomer

Attribute	Datatype	Primary Key		Foreign Key		Referential Integrity				Additional Comments
		Yes	No	Yes	No	Parent		On Update	On Delete	
						Table	Attribute			
Orderno	VarChar	Yes			No					
totalAmount	VarChar		No		No					
Tax	VarChar		No		No					
Delstreet	VarChar		No		No		Deladd			
DelCity	VarChar		No		No		Deladd			
DelState	VarChar		No		No		Deladd			
DelZip	VarChar		No		No		Deladd			
DelAreaCode	int		No		No		Deladd			
DelNumber	char		No		No		Deladd			
CustId	char			yes		Customer		Cascade	Cacade	

Table 3. Definition of Table: Supplier

Attribute	Datatype	Primary Key		Foreign Key		Referential Integrity				Additional Comments
		Yes	No	Yes	No	Parent		On Update	On Delete	
						Table	Attribute			
supplierNo	char	Yes			No					
supName	VarChar		No		No					
countryCode	int		No		No		Supadd			
areaCode	int		No		No		Supadd			
phnumber	char		No		No					
street	varchar		No		No		Supadd			
city	Varchar		No		No		Supadd			
state	Varchar		No		No		Supadd			
zipPostalCode	Varchar		No		No		Supadd			
Country	Varchar		No		No		Supadd			
contactName	Varchar		No		No					

Table 4. Definition of Table: Item

Attribute	Datatype	Primary Key		Foreign Key		Referential Integrity				Additional Comments
		Yes	No	Yes	No	Parent		On Update	On Delete	
						Table	Attribute			
itemNo	char	Yes			No					
itemName	VarChar		No		No					
unitPrice	int		No		No					
qtyOnHand	int		No		No					
reorderPoint	VarChar		No		No					

Table 5. Definition of Table: employees

Attribute	Datatype	Primary Key		Foreign Key		Referential Integrity				Additional Comments
		Yes	No	Yes	No	Parent		On Update	On Delete	
						Table	Attribute			
empld	char	Yes			No					
EmpName	varChar		No		No					
street	varchar		No	Yes	No		Address			
City	varchar		No		No		Address			
State	varchar		No		No		Address			
orderNo	varchar		no	Yes		orderofcuster		cascade	cascade	

Table 6. Definition of Table: Contains

Attribute	Datatype	Primary Key		Foreign Key		Referential Integrity				Additional Comments
		Yes	No	Yes	No	Parent		On Update	On Delete	
						Table	Attribute			
Orderno	Char	Yes		Yes		Orderofc stomer		Cascade	Cascade	
ItemNo	Char	yes		Yes		Item		Cascade	Cascade	
Quantity	Int		No		No					

Table 7. Definition of Table: Supplies

Attribute	Datatype	Primary Key		Foreign Key		Referential Integrity				Additional Comments
		Yes	No	Yes	No	Parent		On Update	On Delete	
						Table	Attribute			
SupplierNo	Char	Yes		Yes		Orderofcustomer		Cascade	Cascade	
ItemNo	Char	yes		Yes		Item		Cascade	Cascade	
UnitPrice	int		No		No					

Script:

```
create database Onlinestore;
```

```
-- -----Select the database for use-----
use OnlineStore;
```

```
-- -----Create the tables-----
```

```
CREATE TABLE Customer (
    Custid CHAR(4) NOT NULL,
    firstName VARCHAR(25) NOT NULL DEFAULT "",
    lastName VARCHAR(25) NOT NULL DEFAULT "",
    street VARCHAR(7) NOT NULL DEFAULT "",
    city VARCHAR(12) NOT NULL DEFAULT "",
    state VARCHAR(12) NOT NULL DEFAULT "",
    zip CHAR(6) NOT NULL DEFAULT "",
    creditlimit int,
    areaCode CHAR(6) NOT NULL DEFAULT "",
    Phnumber CHAR(10) NOT NULL DEFAULT "",
    PRIMARY KEY (Custid),
    index (custid)
);
insert into Customer(custId, firstName, lastName, street, city, state, zip, creditLimit,
areaCode,phnumber)
values ('1235', 'Jack', 'Knife', 'Scott', 'Surrey', 'BC', '823993', 2000, 'UDICOD',
'9382933884'),
('3321', 'Garry', 'Box', 'Lonix', 'Regina', 'SK', '422233', 2300, 'UCDCOD', '8899954543'),
('2112', 'Harry', 'Lame', 'Janix', 'Toronto',
'Ont', '342233', 2500, 'UCDC23', '8892355543'),
('3223', 'Garry', 'Jolly', 'Tazin', 'Regina', 'SK', '425533', 9800, 'CUICOD', '889748934'),
('1243', 'Brory', 'Luke', 'Broxy', 'Monkina',
'BRUN', '568833', 3200, 'UHDOOD', '8899983943');
```

```

CREATE TABLE Orderofcustomer (
  orderNo varchar(3) NOT NULL,
  TotalAmount varchar(6) not null default "",
  Tax varchar(4) not null default "",
  Delstreet VARCHAR(18) not null default "",
  DelCity VARCHAR(18) not null default "",
  DelState VARCHAR(15) NOT NULL DEFAULT "",
  DelZip varchar(10) NOT NULL DEFAULT "",
  DelAreaCode CHAR(6) NULL DEFAULT "",
  DelNumber char(12) NOT NULL DEFAULT "",
  Custid CHAR(4) NOT NULL,
  PRIMARY KEY (OrderNo),
  FOREIGN KEY (Custid)
    REFERENCES Customer (custid)
    ON DELETE CASCADE ON UPDATE CASCADE
);

insert into Orderofcustomer (orderNo, totalAmount, tax, DelStreet, DelCity, DelState,
DelZip, DelAreaCode, DelNumber, custId)
values      ('971', '99', '49', 'Scottdale',
'Surrey', 'BC', '568833', 'UHDOOD', '8829383989', '1235' ),
            ('972', '100', '8', 'Rexin', 'Halifax'
, 'BRUN', '568833', 'UHDOOD', '88973983943', '3321'),
            ('973', '200', '30', 'Blackmount', 'hollock',
'BRUN', '568833', 'UHDOOD', '628392839', '2112'),
            ('974', '324', '38', 'gorita', 'Regina',
'SK', '422233', 'UCDCOD', '8899954543', '3223' ),
            ('975', '544', '40', 'Monkina', 'York',
'BRUN', '568833', 'UHDOOD', '8899983943', '1243' );

SHOW tables;
SHOW databases;
DESCRIBE Customer;
describe Orderofcustomer;

```

```

CREATE TABLE Supplier (
  supplierNo CHAR(4) NOT NULL,
  supname VARCHAR(25) NOT NULL DEFAULT "",
  countryCode int,
  areaCode int,
  phnumber char(10) NOT NULL DEFAULT "",
  street varchar(50) NOT NULL DEFAULT "",
  city varchar(50) NOT NULL DEFAULT "",
  state varchar(15) NOT NULL DEFAULT "",
  zipPostalCode varchar(10) NOT NULL DEFAULT "",
  Country varchar(12) NOT NULL DEFAULT "",
  contactName varchar(15) NOT NULL DEFAULT "",

```

primary key (supplierNo),
index (supplierNo)

);

insert into Supplier(supplierNo, supName, countryCode, areaCode , phnumber, street, city, state, zipPostalCode, Country, contactName)

values ('O443', 'Raymond', '91', '433233', '3233432233', '34
phonix', 'Mirzapur', 'UP', '140293', 'India', 'rayind'),
('O444', 'Bal', '92', '968233', '5739049094', '554
pindalaadha', 'Ganga', 'HAR', 'h73h33', 'Pakistan', 'BalPak'),
('O445', 'Gucci', '1', '223344', '9043444333', '344
apnasiritem', 'chandi', 'PUN', 'v23213', 'Canada', 'GucCan'),
('O446', 'Adi', '1', '653343', '2353334443', '22 Padaleyakrkuj', 'San
fransico', 'NAG', '193833', 'USA', 'Adiusa'),
('O447', 'Nike', '8', '834334', '9804934333', '233
vehliKhanaGijyaVA', 'chimhi', 'AP', '8jeihi', 'China', 'NiCHI');

show create table Orderofcustomer;

CREATE TABLE item (

itemNo CHAR(4) NOT NULL,
itemName varchar(15) not null,
UnitPrice int,
qtyOnHand int,
ReorderPoint varCHAR(15) NOT NULL,
PRIMARY KEY (itemNo),
Index (itemNo),
Index (UnitPrice)

);

insert into item(itemNo, itemName, unitPrice, qtyOnHand, reorderPoint)

values ('1005', 'lphone', 1300, 1, 'DAU'),
('1006', 'Galaxym10', 799, 5, 'leadtime'),
('1007', 'Dell TV', '1235', 64, 'safestock'),
('1008', 'Gucci pant', 3223, 65, 'DAU'),
('1009', 'Bal shirt', 124, 15, 'Safestock');

CREATE TABLE employees (

empld CHAR(4) NOT NULL,
EmpName varchar(15) not null,
Street varchar(15) not null,
city varchar(15) not null,
State varchar(15) not null,
orderNo varchar(3) NOT NULL,
PRIMARY KEY (empid),
FOREIGN KEY (orderNo)
REFERENCES orderofcustomer (orderNo)

```
ON DELETE CASCADE ON UPDATE CASCADE,
Index (empld)
);
insert into employees(empld, EmpName, street, city, State, orderNo)
values ('2001', 'Prabhjot', 'Tipusultan', 'Mogiabaad','Bihar','971' ),
      ('2002', 'Jassa', '125 str', 'Surrey','BC','972' ),
      ('2003', 'Japjot', '92 ave', 'Burnaby','BC','973' ),
      ('2004', 'Jot', '32 str', 'Vancouv','BC','974' ),
      ('2005', 'Jason', '198 str', 'Calgary','AB','975' );
```

```
CREATE TABLE supplies (
supplierNo CHAR(4) NOT NULL,
itemNo CHAR(4) NOT NULL,
UnitPrice int,
PRIMARY KEY (SupplierNo,itemNo),
Foreign key ( UnitPrice)
References item (UnitPrice)
ON DELETE CASCADE ON UPDATE CASCADE,
index (Unitprice),
index(supplierNo)
);
insert into supplies(SupplierNo,itemNo,UnitPrice)
values ('0443','1005',1300),
      ('0444','1006',799),
      ('0445','1007',1235),
      ('0446','1008',3223),
      ('0447','1009',124);
```

```
CREATE TABLE Contains (
orderno CHAR(4) NOT NULL,
ItemNo CHAR(4) NOT NULL,
Quantity int,
PRIMARY KEY (orderNo,itemNo),
FOREIGN KEY (orderNo)
REFERENCES orderofCustomer (orderNo)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (itemNo)
REFERENCES item (itemNo)
ON DELETE CASCADE ON UPDATE CASCADE
);
```

```
insert into Contains(orderNo,ItemNo,Quantity)
values ('971','1005',2),
      ('972','1006',3),
      ('973','1007',2),
      ('974','1008',3),
      ('975','1009',1);
```

show tables;

```
SELECT * FROM supplier;  
SELECT * FROM Customer;  
SELECT * FROM Orderofcustomer;  
SELECT * FROM supplies;  
SELECT * FROM item;  
SELECT * FROM Contains;
```

drop database onlinestore;

3. USING THE PRODUCT

3.1. Special considerations

Even though the product has been meticulously designed and user-friendly, certain aspects must be handled with greater care. First, try not to run too many instructions at once; the system must process each one at a time, which lessens the stress on RAM and makes the processor more durable. Furthermore, ensure that the user reboots the system on time, as this boosts its speed and, as a result, reduces mental strain and improves the system's reliability. Finally, the information provided by the superuser is crucial since it contains personal information. As a result, it should be encrypted and protected by more secure networks and software to prevent data breaches.

3.2. Instructions for use

A database management system (DBMS) provides several approaches for storing and retrieving data. Furthermore, DBMS serves as an efficient handler to balance the needs of multiple applications that use the same Uniform data management techniques. As a result, application programmers are never exposed to data representation and storage details. Instead, a DBMS employs a variety of powerful functions to effectively store and retrieve data while maintaining data integrity and security. Therefore, access to the database is only shared with an authorized person. The database commands are also used to filter the precise information about the user. It changes the specific information about the customer by using alter, changes and updates command. The values should be recorded correctly as part of the instructions since changing the database would be possible. Schema's tables are all linked together. The presence of foreign keys will obligate the process. In other words, disrupting or changing values in one table may impact all tables.