

Database Project Document

Batoor Delivery Services

Submitted by

Saifullah Amin, Maazullah, Shahid Ali & Muhammad Junaid

Submitted to

Muhammad Yousaf

Date

15th August, 2020

Abstract

The web application is to be made for Batoor Delivery Services, Swabi. This will be used by their customers to place on-demand orders and do online shopping from the registered restaurants. The admin panel will have an option to approve/disapprove orders and get a display of the orders placed. They can also add/edit items & restaurants.

Contents

CHAPTER 1 INTRODUCTION	
PROBLEM STATEMENT	
INTRODUCTION (OVERVIEW)	
SCOPE	
USERS.....	
FUNCTIONAL REQUIREMENT.....	
NON FUNCTIONAL REQUIREMENT	
SOFTWARE / TOOLS / PROGRAMMING LANGUAGES USED	
CHAPTER 2 DATABASE DESIGN PROCESS	
CONCEPTUAL DESIGN	
LOGICAL DESIGN	
1NF (FIRST NORMAL FORM)	
2NF (SECOND NORMAL FORM).....	
3NF (THIRD NORMAL FORM)	
CHAPTER 3 DATABASE IMPLEMENTATION	
DATABASE CREATION QUERIES	
TABLES IMPLEMENTATION SCREENSHOTS.....	

Problem Statement

Title:

Automating the manual ordering system of Batoor Delivery Services, Swabi and making it easy-to-use for everyone through internet.

Description:

Currently, Batoor Delivery Services have a manual system of taking orders from their customers. Customers call them on their mobile number and place their order. They note all their details on paper and make a record of it. The customers do not have any visual of the restaurants and food they offer.

If the system gets digitalized and have a website. The user will have a visual of restaurants listing and the meals they offer. They will be able to select anything of their choice. They will also be able to make on-demand orders online without the process of making a phone call. The paper-based records will also be replaced with the data of each order stored in the database.

Introduction (Overview)

The project is of a website for Batoor Delivery Services. The website will be used by the company to interact with their customers. The users can register/login to the website for using its services. The website will have a listing of restaurants that are registered with the company and the food they offer. The users will browse the items and add any of them to cart. The users will have an option to increase/decrease the quantity of items. The users upon completion his order will checkout and enter drop-off address. The users also can place on-demand orders for different products. They will input details about the product and pickup/drop-off locations and will place the order. The orders upon placement will be received by the admin for approval/disapproval. The admin will input delivery charges for online shopping and delivery charges plus products price for on-demand delivery prior to approval. Upon approval a text message will be sent to the customer's contact about his order approval. The admin panel upon login will have options to add/edit restaurants and items for display on the website.

The website will be developed using MERN Stack Technology. HTML, CSS, JavaScript, Bootstrap and React.js will be used in front-end. JavaScript, Node.js and Express.js will be used in the backend and NoSQL database MongoDB will be used in the database. For database design process we will use app.diagrams.net online tool. Database implementation just for this task/document will be done using MySQL RDBMS.

Scope

This document applies to Batoor Delivery Services web app. This web app facilitates the customers of the company to place online on-demand orders and do online shopping from their registered restaurants. The admin will have a display of two portions: orders to be approved and orders approved. They can add/edit items and restaurants as well.

The web app has two panels of administration and users. They can access it by providing login credentials. The output then comprises of an interactive display that let the users select the desirable function that he wants to perform.

The option of online payment would not be available by now and would be added later. The web app can only be accessed through internet.

The software is expected to be completed in 3 months approximately.

Users

- **Admin:** The admin of the web app will have a display of order to approve and approved orders. He can approve / not approve the orders. The approved orders will be stored, and the admin can access it any time. The admin will have an option to add/edit items and restaurants. These functionalities will be displayed categorically with a good interface.
- **User / Customer:** The user / customer will have the option to make on-demand order online or do online shopping from selected restaurants. The user will have the option to login / create account. These functionalities will be displayed categorically with a good interface.

Functional Requirement

- **Orders Display**
 - The web app will display to-be-approved and approved orders to the admin
 - The web app would allow the admin panel to send confirmation messages to the user for their orders with delivery charges.
 - The web app should display the records of all approved orders to the admin.
 - The admin should have an option of submitting delivery charges prior to approving online shopping order.

- The admin should have option of submitting price & delivery charges for on-demand delivery after approval.
- **Add/Edit Items and Restaurants**
 - The web app should allow the admin to add/edit items for online order.
 - The web app should allow the admin to add/edit restaurants.
- **Login / Create Account**
 - The web app shall allow the admin to login with the credentials.
 - The web app shall allow the customer to login to the website.
 - The web app shall allow the customer to create account for the website.
- **On-Demand Delivery**
 - The web app shall allow the customer to place on-demand delivery orders.
 - The order will require name, contact, and address of pickup and drop-off locations.
 - The web app shall allow the customer to state the product the customer wants to order.
- **Online Shopping (Restaurants)**
 - The web app shall allow the customer to do online shopping from the selected restaurants.
 - The web app will have a display of all the restaurants and the food offered by them. The user will select the desired meal and add it to the cart. The user at last, will checkout from the cart and make the final order.
- **Search Box**
 - The search box will allow the user to search any of their desired item.
- **Shopping Cart**
 - Shopping Cart will help user to buy multiple items in a single go. User just have to navigate to different products and press Add to Card button on each Product Page.
- **Feedback**
 - This will allow the user to give his reviews or suggestions about the product and services provide by the System.
- **Technical Issues**
 - This system will work on client-server architecture. It will require an internet server, and which will be able to run application. The system should support some commonly used browser such as IE or Google Chrome etc.

- **Validity Checks**

- In order to gain access to the system, the user is required to enter his/her correct username and password. Also, if the user is an administrator, he is required to enter his login id in order to access and change the facilities provided by the system.

Non-Functional Requirement

Performance Requirements

The following list provides a brief summary of the performance requirements for this software

- **Capacity:** The software should be available to the audience for 24 hours service. The software should be able to operate on all major web-browsers with all of its fundamental functions. It should not slow-down the system. Even at peak hours without affecting the quality of service of the system
- **Dynamic requirements:**
 - The website opening load should not exceed from 2 to 3 sec in normal and under the heavy load not more than 5 to 6 sec.
 - The login credential verification time must not exceed 1 sec. under normal server workload and 3 sec. under peak server workload.
 - Creating user account should not exceed 1 sec in normal workload and 2 sec in heavy workload.
 - Sending approval SMS from admin to user must not exceed 1 sec.
- **Quality:** The goal is to produce quality software. As the quality of a software is difficult to measure quantitatively, the following ways or guidelines will be used when judging the quality of the software:
 1. **Consistency** – All code will be consistent with respect to the style (this is implied when adhering to the standard).
 2. **Test Cases** – All functionality will be tested thoroughly.

Software System Attributes

- **Availability:** The software will be available 24 hours a day and 7 day per week for its users having internet connection. In case of any problem accessing the application, the support team will solve the problem as soon as possible.

- **Maintainability:** The system should be maintainable, any changes required should be applicable on the system at any time without any increase in errors.

Security Requirements:

The system should provide a secure login to the users by using advanced secure login algorithms and provide access only to the authorized users as security is the key requirement of this system. The password shall be 6-14 characters long. Passwords can contain digit, hyphen and underscore.

The system's back-end servers shall never display a customer's password. The customer's password may be reset but never shown. The system's back-end servers shall only be accessible to authenticated administrators. The system's back-end databases shall be encrypted.

5.4 Software Quality Attributes:

System must be:

- Consistent in performance
- Safe and Secure
- Robust
- Scalable
- Flexible
- User friendly
- Efficient
- Inter-operable
- Upgradable

Software / Tools / Programming Languages used

Front-end:

- HTML
- CSS
- JavaScript
- Bootstrap
- React.js

Back-end:

- JavaScript
- Node.js
- Express.js

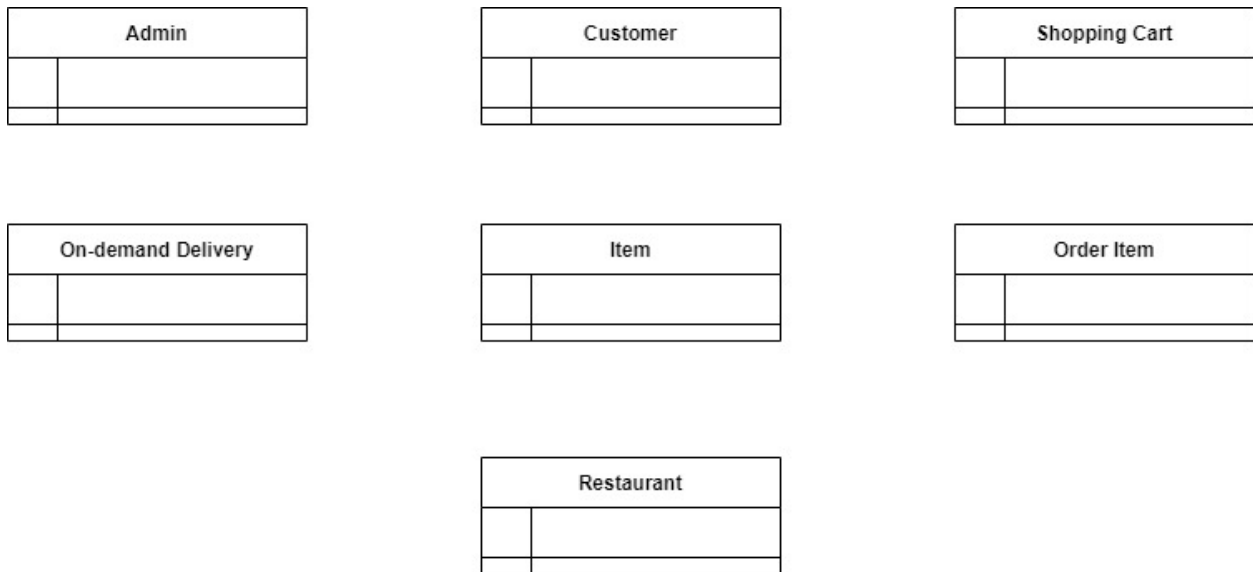
Database:

- MongoDB
- MySQL (for this document only)

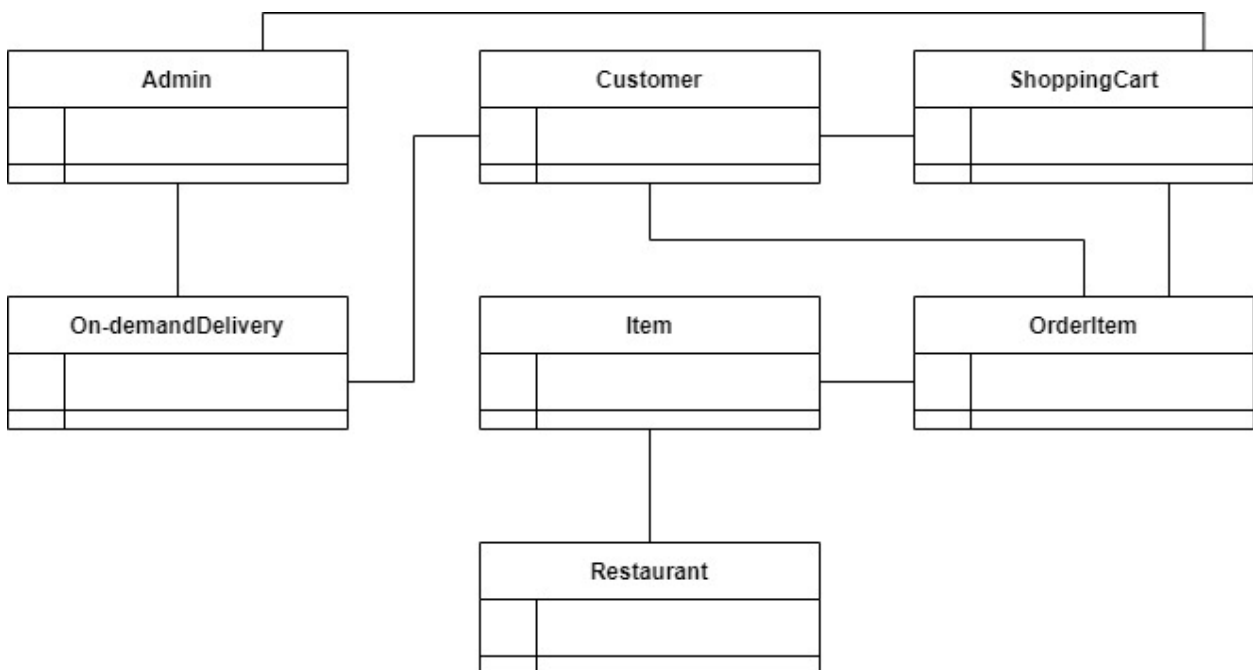
Chapter 2 Database Design Process

Conceptual Design

Step 1:

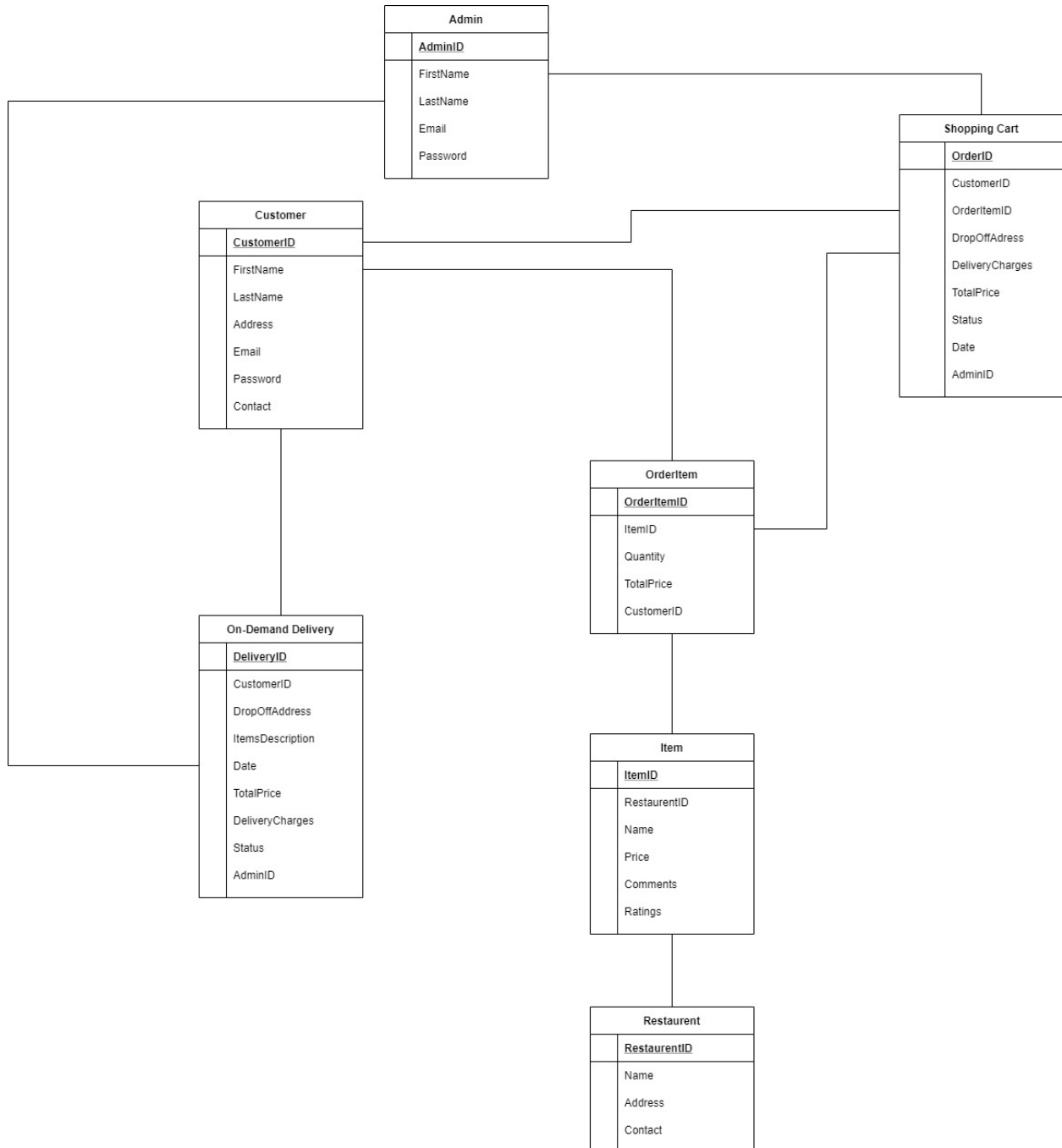


Step 2:

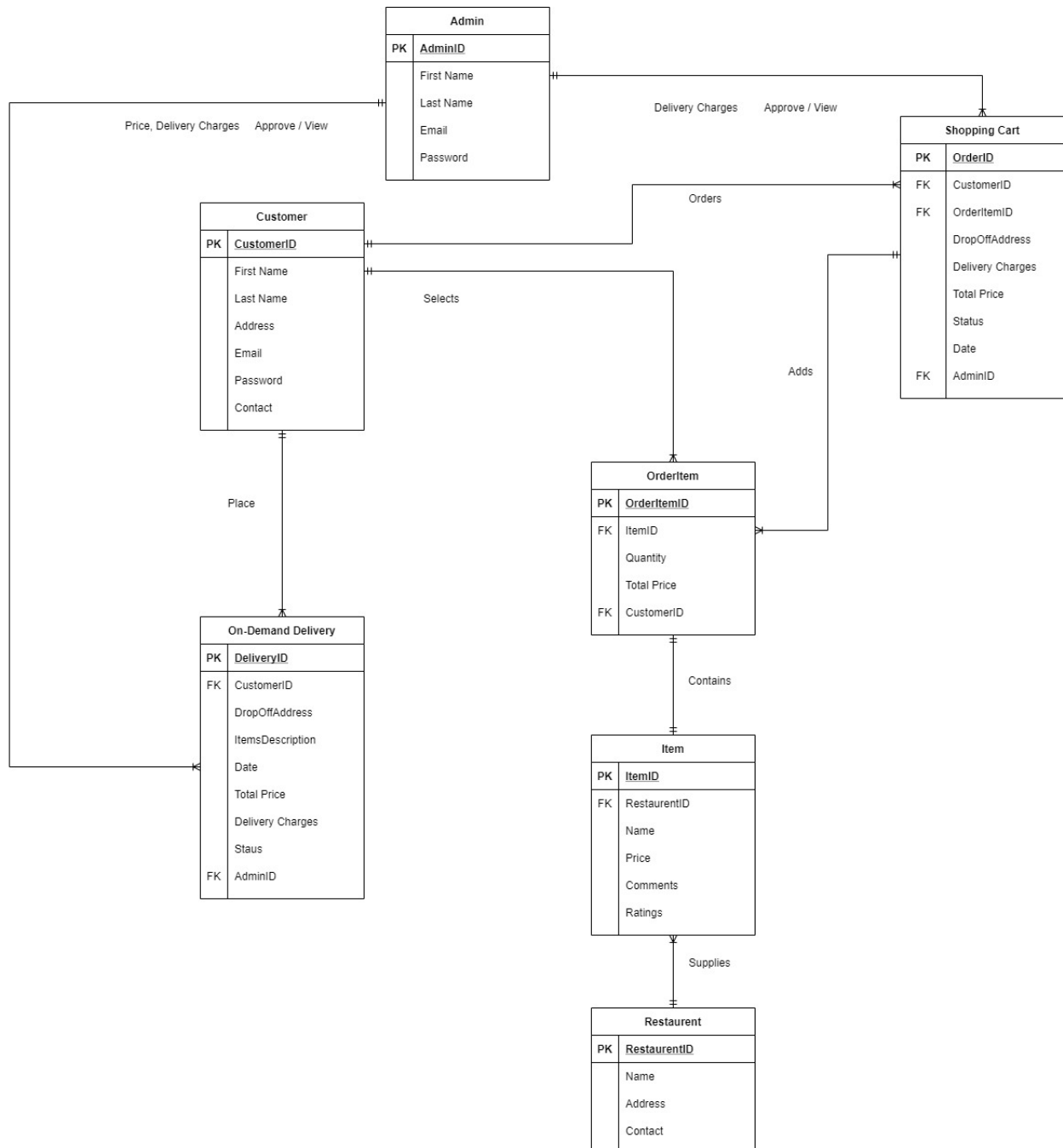


Logical Design

Step 1:

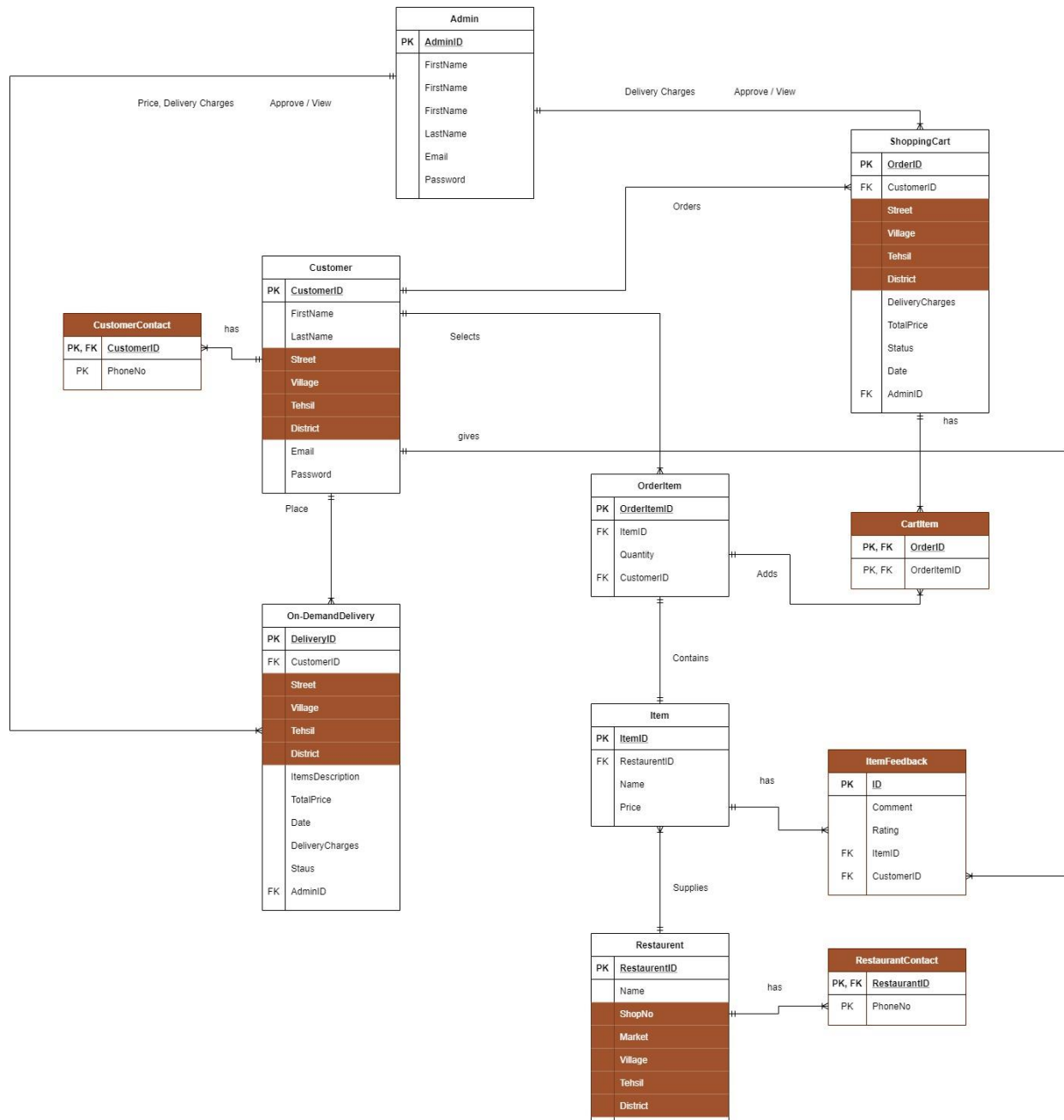


Step 2:



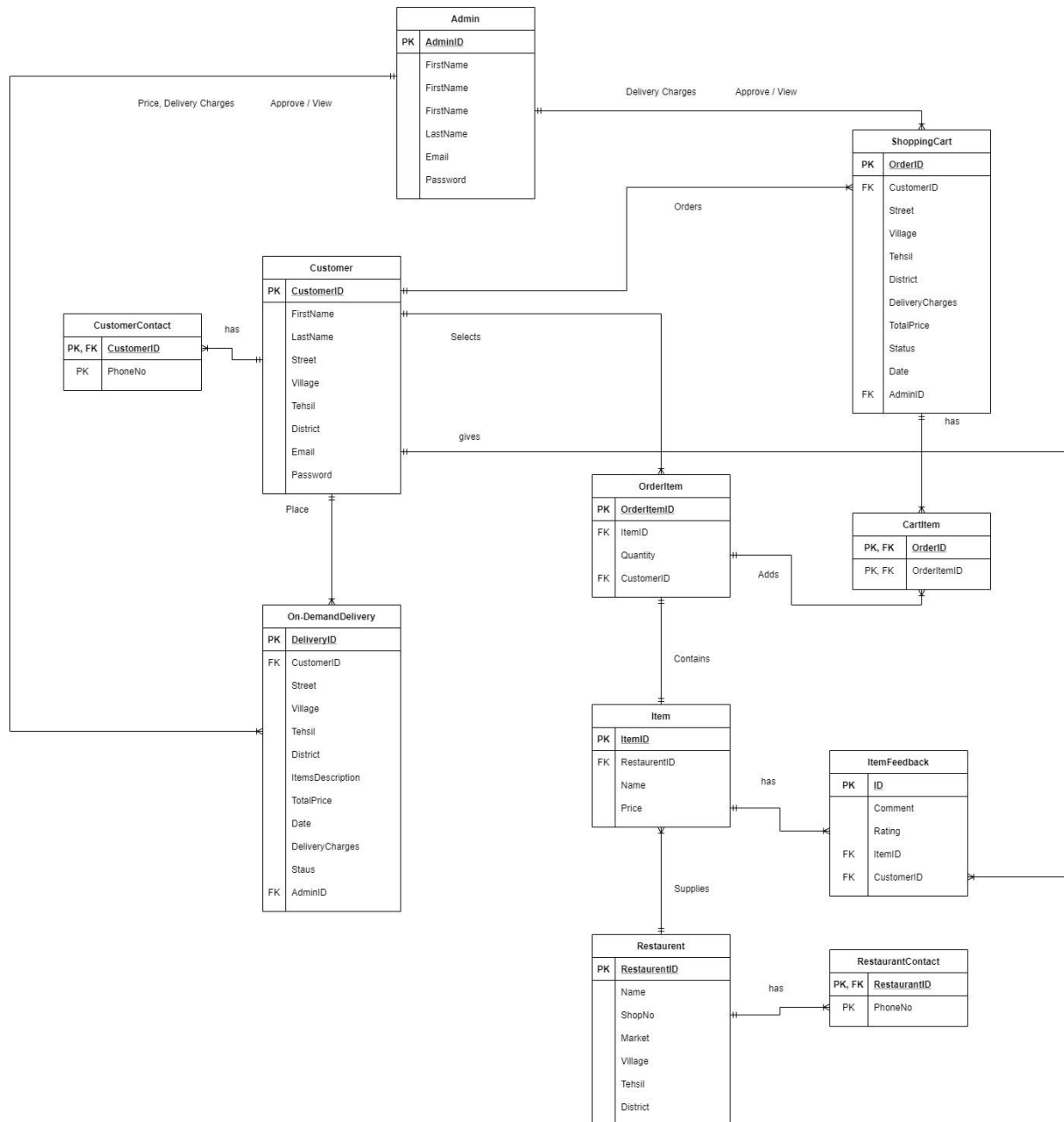
1NF (First Normal Form)

(Changes highlighted)



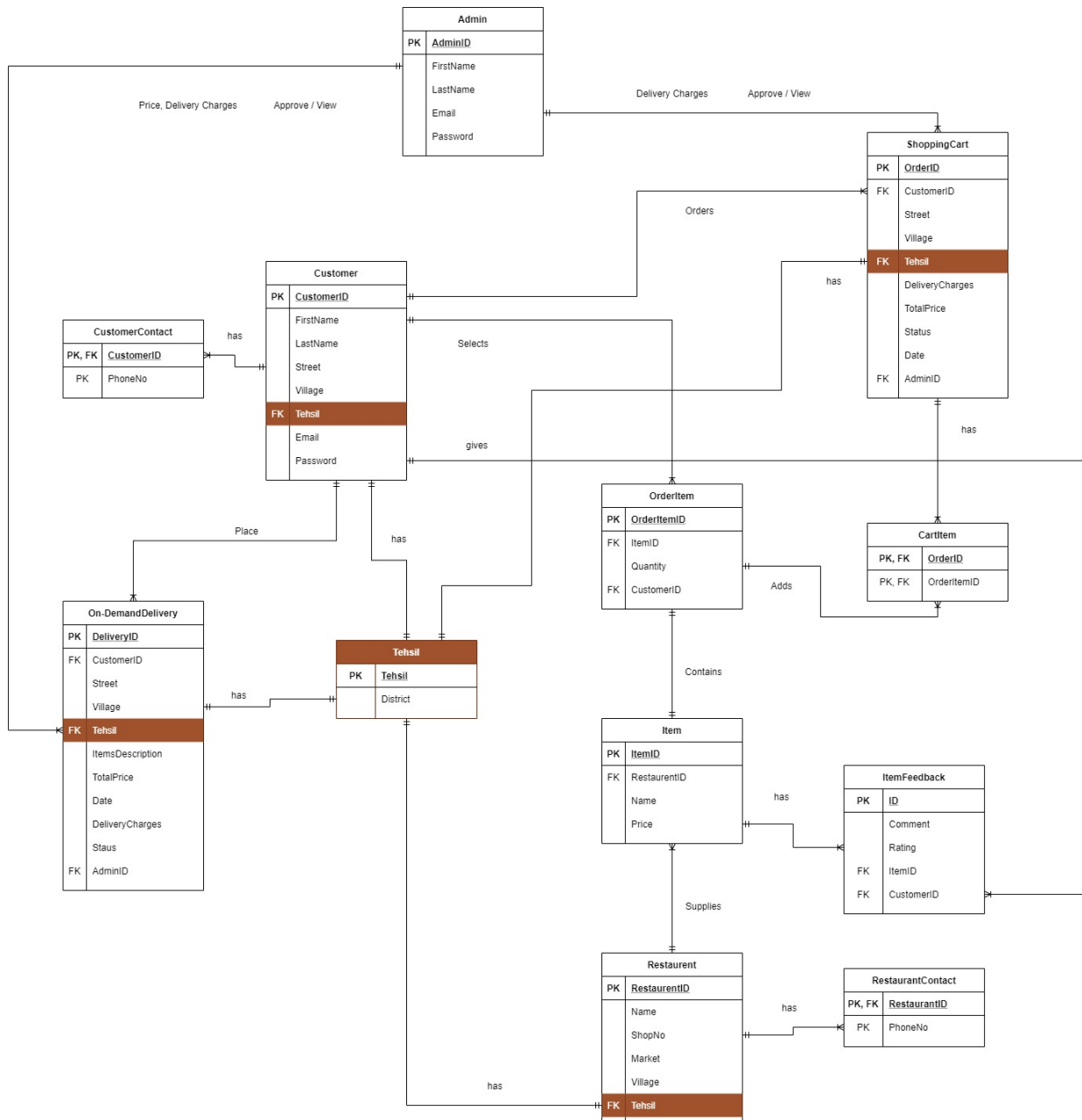
2NF (Second Normal Form)

(No Changes)

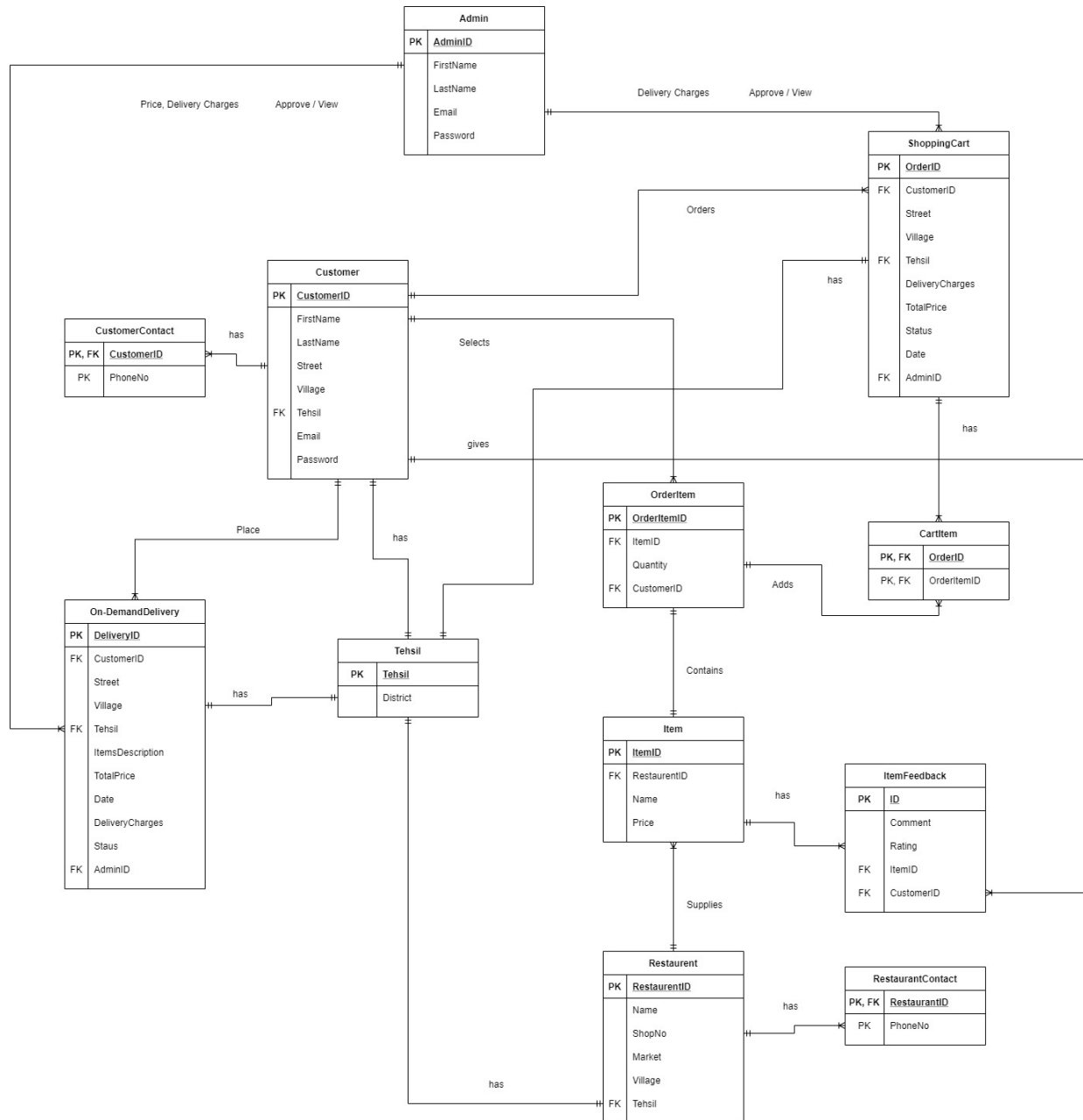


3NF (Third Normal Form)

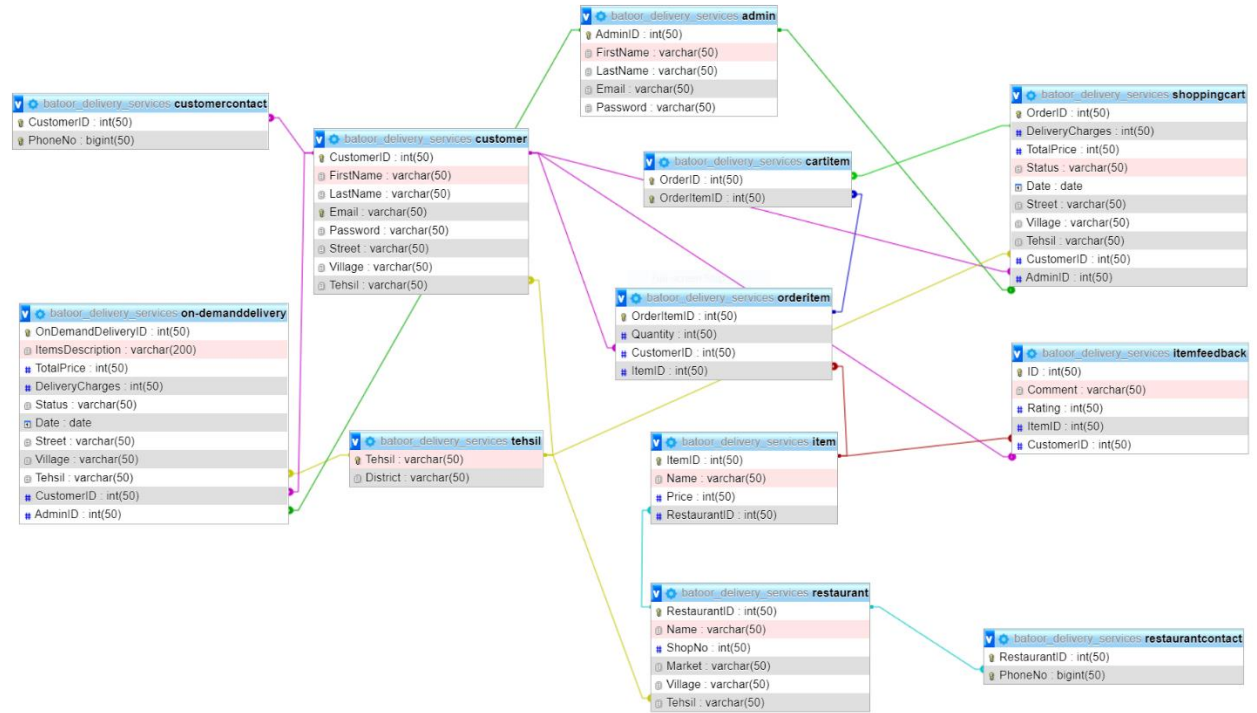
(Changes Highlighted)



Final ERD



Final ERD on MySQL



Chapter 3 Database Implementation

Database Creation Queries

Admin Table

```
CREATE TABLE `admin` (  
    `AdminID` int(50) NOT NULL AUTO_INCREMENT,  
    `FirstName` varchar(50) NOT NULL,  
    `LastName` varchar(50) NOT NULL,  
    `Email` varchar(50) NOT NULL,  
    `Password` varchar(50) NOT NULL,  
    PRIMARY KEY (`AdminID`)  
)
```

Customer Table

```
CREATE TABLE `customer` (  
    `CustomerID` int(50) NOT NULL AUTO_INCREMENT,  
    `FirstName` varchar(50) NOT NULL,  
    `LastName` varchar(50) NOT NULL,  
    `Email` varchar(50) NOT NULL,  
    `Password` varchar(50) NOT NULL,  
    `Street` varchar(50) NOT NULL,  
    `Village` varchar(50) NOT NULL,  
    `Tehsil` varchar(50) NOT NULL,  
    PRIMARY KEY (`CustomerID`),  
    UNIQUE KEY `Email` (`Email`),  
    KEY `Tehsil` (`Tehsil`),  
    CONSTRAINT `customer_ibfk_1` FOREIGN KEY (`Tehsil`) REFERENCES `tehsil` (`Tehsil`)  
)
```

CustomerContact Table

```
CREATE TABLE `customercontact` (  
    `CustomerID` int(50) NOT NULL,  
    `PhoneNo` bigint(50) NOT NULL,  
    PRIMARY KEY (`CustomerID`,`PhoneNo`),  
    CONSTRAINT `customercontact_ibfk_1` FOREIGN KEY (`CustomerID`) REFERENCES  
    `customer` (`CustomerID`)  
)
```

On-DemandDeilvery Table

```
CREATE TABLE `ondemanddelivery` (  
    `OnDemandDeliveryID` int(50) NOT NULL AUTO_INCREMENT,  
    `ItemsDescription` varchar(200) NOT NULL,  
    `TotalPrice` int(50) DEFAULT NULL,  
    `DeliveryCharges` int(50) DEFAULT NULL,  
    `Status` varchar(50) DEFAULT NULL,  
    `Date` date NOT NULL,  
    `Street` varchar(50) NOT NULL,  
    `Village` varchar(50) NOT NULL,  
    `Tehsil` varchar(50) NOT NULL,  
    `CustomerID` int(50) NOT NULL,  
    `AdminID` int(50) DEFAULT NULL,  
    PRIMARY KEY (`OnDemandDeliveryID`),  
    KEY `CustomerID` (`CustomerID`),  
    KEY `Tehsil` (`Tehsil`),  
    KEY `AdminID` (`AdminID`),  
    CONSTRAINT `ondemanddelivery_ibfk_1` FOREIGN KEY (`Tehsil`) REFERENCES `tehsil`  
    (`Tehsil`),
```

```
CONSTRAINT `ondemanddelivery_ibfk_2` FOREIGN KEY (`CustomerID`) REFERENCES
`customer` (`CustomerID`),

CONSTRAINT `ondemanddelivery_ibfk_3` FOREIGN KEY (`AdminID`) REFERENCES `admin`
(`AdminID`)

)
```

Restaurarant Table

```
CREATE TABLE `restaurant` (
  `RestaurantID` int(50) NOT NULL AUTO_INCREMENT,
  `Name` varchar(50) NOT NULL,
  `ShopNo` int(50) NOT NULL,
  `Market` varchar(50) NOT NULL,
  `Village` varchar(50) NOT NULL,
  `Tehsil` varchar(50) NOT NULL,
  PRIMARY KEY (`RestaurantID`),
  KEY `Tehsil` (`Tehsil`),
  CONSTRAINT `restaurant_ibfk_1` FOREIGN KEY (`Tehsil`) REFERENCES `tehsil` (`Tehsil`)
)
```

RestaurantContact Table

```
CREATE TABLE `restaurantcontact` (
  `RestaurantID` int(50) NOT NULL,
  `PhoneNo` bigint(50) NOT NULL,
  PRIMARY KEY (`RestaurantID`,`PhoneNo`),
  CONSTRAINT `restaurantcontact_ibfk_1` FOREIGN KEY (`RestaurantID`) REFERENCES
`restaurant` (`RestaurantID`)
)
```

Item Table

```
CREATE TABLE `item` (  
  `ItemID` int(50) NOT NULL AUTO_INCREMENT,  
  `Name` varchar(50) NOT NULL,  
  `Price` int(50) NOT NULL,  
  `RestaurantID` int(50) NOT NULL,  
  PRIMARY KEY (`ItemID`),  
  KEY `RestaurantID` (`RestaurantID`),  
  CONSTRAINT `item_ibfk_1` FOREIGN KEY (`RestaurantID`) REFERENCES `restaurant`  
  (`RestaurantID`)  
)
```

ItemFeedback Table

```
CREATE TABLE `itemfeedback` (  
  `ID` int(50) NOT NULL AUTO_INCREMENT,  
  `Comment` varchar(50) NOT NULL,  
  `Rating` int(50) NOT NULL,  
  `ItemID` int(50) NOT NULL,  
  `CustomerID` int(50) NOT NULL,  
  PRIMARY KEY (`ID`),  
  KEY `ItemID` (`ItemID`),  
  KEY `CustomerID` (`CustomerID`),  
  CONSTRAINT `itemfeedback_ibfk_1` FOREIGN KEY (`CustomerID`) REFERENCES `customer`  
  (`CustomerID`),  
  CONSTRAINT `itemfeedback_ibfk_2` FOREIGN KEY (`ItemID`) REFERENCES `item` (`ItemID`)  
)
```

OrderItem Table

```
CREATE TABLE `orderitem` (  
    `OrderItemID` int(50) NOT NULL AUTO_INCREMENT,  
    `Quantity` int(50) NOT NULL,  
    `CustomerID` int(50) NOT NULL,  
    `ItemID` int(50) NOT NULL,  
    PRIMARY KEY (`OrderItemID`),  
    KEY `ItemID` (`ItemID`),  
    KEY `CustomerID` (`CustomerID`),  
    CONSTRAINT `orderitem_ibfk_1` FOREIGN KEY (`ItemID`) REFERENCES `item` (`ItemID`),  
    CONSTRAINT `orderitem_ibfk_2` FOREIGN KEY (`CustomerID`) REFERENCES `customer`  
    (`CustomerID`)  
)
```

CartItem Table

```
CREATE TABLE `cartitem` (  
    `OrderID` int(50) NOT NULL,  
    `OrderItemID` int(50) NOT NULL,  
    PRIMARY KEY (`OrderID`, `OrderItemID`),  
    KEY `OrderItemID` (`OrderItemID`),  
    CONSTRAINT `cartitem_ibfk_1` FOREIGN KEY (`OrderID`) REFERENCES `shoppingcart`  
    (`OrderID`),  
    CONSTRAINT `cartitem_ibfk_2` FOREIGN KEY (`OrderItemID`) REFERENCES `orderitem`  
    (`OrderItemID`)  
)
```

ShoppingCart Table







```
CREATE TABLE `shoppingcart` (  
    `OrderID` int(50) NOT NULL AUTO_INCREMENT,  
    `DeliveryCharges` int(50) DEFAULT NULL,  
    `TotalPrice` int(50) NOT NULL,  
    `Status` varchar(50) DEFAULT NULL,  
    `Date` date NOT NULL,  
    `Street` varchar(50) NOT NULL,  
    `Village` varchar(50) NOT NULL,  
    `Tehsil` varchar(50) NOT NULL,  
    `CustomerID` int(50) NOT NULL,  
    `AdminID` int(50) DEFAULT NULL,  
    PRIMARY KEY (`OrderID`),  
    KEY `CustomerID` (`CustomerID`),  
    KEY `AdminID` (`AdminID`),  
    KEY `Tehsil` (`Tehsil`),  
    CONSTRAINT `shoppingcart_ibfk_1` FOREIGN KEY (`CustomerID`) REFERENCES `customer`  
    (`CustomerID`),  
    CONSTRAINT `shoppingcart_ibfk_2` FOREIGN KEY (`AdminID`) REFERENCES `admin`  
    (`AdminID`),  
    CONSTRAINT `shoppingcart_ibfk_3` FOREIGN KEY (`Tehsil`) REFERENCES `tehsil` (`Tehsil`)  
)
```

Tehsil Table








```
CREATE TABLE `tehsil` (  
    `Tehsil` varchar(50) NOT NULL,  
    `District` varchar(50) NOT NULL,  
    PRIMARY KEY (`Tehsil`)  
)
```

Tables Implementation Screenshots



















Admin

		AdminID	FirstName	LastName	Email	Password
<input type="checkbox"/>	 Edit  Copy  Delete	1	Saifullah	Amin	saifullah@gmail.com	a7d5a2a2769832c4acc332786db0fb62a1d22a0c209535fe69
<input type="checkbox"/>	 Edit  Copy  Delete	2	Aslam	Khan	aslam@gmail.com	3ac71bc849203c4f4e675736b330c151d872cfaa6dac9796e8

Customer

		CustomerID	FirstName	LastName	Email	Password	Street	Village	Tehsil
<input type="checkbox"/>	 Edit  Copy  Delete	1	Muhammad	Saad	saad@gmail.com	7c1a4a1d4b462cb7b767e6d653440877683f11b5c1b069717c	Yousafzai	Jalsai Dhok	Lahor
<input type="checkbox"/>	 Edit  Copy  Delete	2	Muhammad	Kamran	kamran@gmail.com	d709631b344bfb8b5d4c37f763b53c07d3c3cfdbdaec9eae90	Karimabad	Tordher	Lahor
<input type="checkbox"/>	 Edit  Copy  Delete	3	Ali	Abbas	ali@gmail.com	94419b99b12c11133a4dfeccc3e17885974beb48f7827c4823	GPO Colony	Kalabat	Topi
<input type="checkbox"/>	 Edit  Copy  Delete	4	Nauman	Khan	nauman@gmail.com	09f6a18d5bd66f7147bc8055f2ff86a5fd314f8953f78ae1f0	Bazar Cham	Dagai	Razar















CustomerContact

		CustomerID	PhoneNo
<input type="checkbox"/>	 Edit  Copy  Delete	1	3234572904
<input type="checkbox"/>	 Edit  Copy  Delete	1	3296753167
<input type="checkbox"/>	 Edit  Copy  Delete	2	3138646864
<input type="checkbox"/>	 Edit  Copy  Delete	3	3234572904
<input type="checkbox"/>	 Edit  Copy  Delete	4	3126743972
<input type="checkbox"/>	 Edit  Copy  Delete	4	3458976423






















On-demandDelivery

		OnDemandDeliveryID	ItemsDescription	TotalPrice	DeliveryCharges	Status	Date	Street	Village	Tehsil	CustomerID	AdminID
<input type="checkbox"/>	 Edit  Copy  Delete	2	2 pound cake from Taj Bakers Swabi	500	100	Approved	2020-08-15	Bazar Cham	Tordher	Lahor	1	2
<input type="checkbox"/>	 Edit  Copy  Delete	3	Haier Washing Machine HWM-851708 from Haier Store ...	NULL	NULL	Disapproved	2020-08-15	Yousafzai	Lahor Rapore	Lahor	2	1

Restaurant

 		RestaurantID	Name	ShopNo	Market	Village	Tehsil
<input type="checkbox"/>  Edit  Copy  Delete		1	Burger Point	23	Dildar Plaza	Dagai	Razar
<input type="checkbox"/>  Edit  Copy  Delete		2	Pizza Night	234	Ali Market	Shahmansoor	Swabi
<input type="checkbox"/>  Edit  Copy  Delete		3	Amir Restaurant	1	Hund Hotels	Anbar	Lahor
<input type="checkbox"/>  Edit  Copy  Delete		4	Foodies Fast Food	31	Topi Bazar	Main Topi	Topi
<input type="checkbox"/>  Edit  Copy  Delete		5	Taj Mahal Restaurant	2	Five Star Plaza	Main Swabi Bazar	Swabi

RestaurantContact

 		RestaurantID	PhoneNo
<input type="checkbox"/>  Edit  Copy  Delete		1	3138646864
<input type="checkbox"/>  Edit  Copy  Delete		1	3296753167
<input type="checkbox"/>  Edit  Copy  Delete		2	3126743972
<input type="checkbox"/>  Edit  Copy  Delete		3	3126743972
<input type="checkbox"/>  Edit  Copy  Delete		3	3296753167
<input type="checkbox"/>  Edit  Copy  Delete		4	340875367
<input type="checkbox"/>  Edit  Copy  Delete		5	3339874372














Item

<div><div><div>←</div><div>T</div><div>→</div></div></div>			ItemID	Name	Price	RestaurantID
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		1	Chicken Burger (Small)	200	1
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		2	Chicken Burger (Medium)	250	1
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		4	Cheese Pizza (Small)	500	1
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		5	Cold Drink (250 ml)	50	1
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		6	Fries (Small)	50	2
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		7	Fries (Large)	100	2
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		8	Chicken Tikka Pizza (Small)	500	2
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		9	Chicken Tikka Pizza (Large)	700	2
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		10	Chicken Karai	500	3
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		11	Mutton Karai	500	5
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		12	Pulao	300	5
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		13	Shawarma	100	4
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div></div>		14	Chicken Roast	500	4














ItemFeedback

<div><div><div>←</div><div>T</div><div>→</div></div></div>			ID	Comment	Rating	ItemID	CustomerID
<div><div><div><div></div></div></div><div><div><div><div></div></div></div><div><div><div>Edit</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Copy</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Delete</div><div><div><div></div><div></div><div></div></div></div></div></div></div></div></div></div>	1	Tasty	5	14	4		
<div><div><div><div></div></div></div><div><div><div><div></div></div></div><div><div><div>Edit</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Copy</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Delete</div><div><div><div></div><div></div><div></div></div></div></div></div></div></div></div></div>	2	Nice but took so much time	3	9	2		
<div><div><div><div></div></div></div><div><div><div><div></div></div></div><div><div><div>Edit</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Copy</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Delete</div><div><div><div></div><div></div><div></div></div></div></div></div></div></div></div></div>	3	Timely delivery	4	14	3		
<div><div><div><div></div></div></div><div><div><div><div></div></div></div><div><div><div>Edit</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Copy</div><div><div><div></div><div></div><div></div></div></div></div><div><div><div>Delete</div><div><div><div></div><div></div><div></div></div></div></div></div></div></div></div></div>	4	Hell of a taste	5	9	1		








OrderItem

		OrderItemID	Quantity	CustomerID	ItemID
<input type="checkbox"/>	 Edit  Copy  Delete	1	2	3	11
<input type="checkbox"/>	 Edit  Copy  Delete	2	1	3	2
<input type="checkbox"/>	 Edit  Copy  Delete	4	1	4	7
<input type="checkbox"/>	 Edit  Copy  Delete	5	2	4	5














CartItem

		OrderID	OrderItemID
<input type="checkbox"/>	 Edit  Copy  Delete	3	1
<input type="checkbox"/>	 Edit  Copy  Delete	3	2
<input type="checkbox"/>	 Edit  Copy  Delete	4	4
<input type="checkbox"/>	 Edit  Copy  Delete	4	5

ShoppingCart

		OrderID	DeliveryCharges	TotalPrice	Status	Date	Street	Village	Tehsil	CustomerID	AdminID
<input type="checkbox"/>	 Edit  Copy  Delete	3	200	1000	Approved	2020-08-15	Bazar Cham	Dagai	Razar	1	1
<input type="checkbox"/>	 Edit  Copy  Delete	4	100	690	Approved	2020-08-15	Akhunzadgan	Anbar	Lahor	4	2

Tehsil

		Tehsil	District
<input type="checkbox"/>	 Edit  Copy  Delete	Lahor	Swabi
<input type="checkbox"/>	 Edit  Copy  Delete	Razar	Swabi
<input type="checkbox"/>	 Edit  Copy  Delete	Swabi	Swabi
<input type="checkbox"/>	 Edit  Copy  Delete	Topi	Swabi