

**Applied Database Systems**

**IT 530**

**Dr. Carol Jim**

Thursday, May 10, 2018

Lavan Gujjula

Chiranjibi Ghimire

Neha Prasad

Table of Contents

1. [Overview 2](#_Toc513739865)

[Topic Selection](#_Toc513739866)

[Project Background and Description](#_Toc513739867)

[Project Scope](#_Toc513739868)

[High-Level Requirements](#_Toc513739869)

[Affected Parties](#_Toc513739870)

1. [Entity Relationship Design 4](#_Toc513739871)

[Requirements](#_Toc513739872)

[Entity Relationship Diagram](#_Toc513739873)

[Assumptions](#_Toc513739874)

1. [Schema and SQL Implementation 8](#_Toc513739875)

[Relational Schema](#_Toc513739876)

[Implementation of the Schema in MariaDB](#_Toc513739877)

[Create User Interface Design:](#_Toc513739878)

[Main Functionality of Each Page:](#_Toc513739879)

# Overview

## Topic Selection

Restaurant Management System

## Project Background and Description

The main goal of this project is to create a restaurant management system for multiple restaurants. It encompasses of an infrastructure built for quick access to capabilities in the management of; staff, menus, bills, orders, etc. along with an exhaustive customer information portal. The goal of this relational database management system is to provide detailed information on day-to-day activities to efficiently run a profitable business as well as forecast future requirements for the management to plan on.

## Project Scope

The restaurant management system is created to efficiently manage a number of restaurants by an owner. The system will contain 15 entities and relationships combined. The front-end of the system is built using HTML, CSS, JavaScript and PHP languages along with MySQL that is used for database management. It is scheduled to be completed on May 10, 2018.

## High-Level Requirements

The new system must include the following:

* Ability to provide demographic details of each restaurant; such as restaurant name, address, phone number, and hours of operation
* Ability to display multiple menu items with the capability of showcasing items on sale
* Ability to display orders given by the customers and their respective bills
* An interface where managers of each restaurants can view information related to it’s staff

**Deliverables**

The deliverables included for this project include:

* Project proposal with scope
* Entity relationship design
* Schema and SQL Implementation
* User interface design
* Presentation and final report

## Affected Parties

The business processes that will be impacted during this process are discussed below. There will be a digital transformation of all information in the database. The business units that would be affected are; staff, bills, orders, menus, customers and restaurant.

Below is the breakdown of all the entities and relationships that will be used:

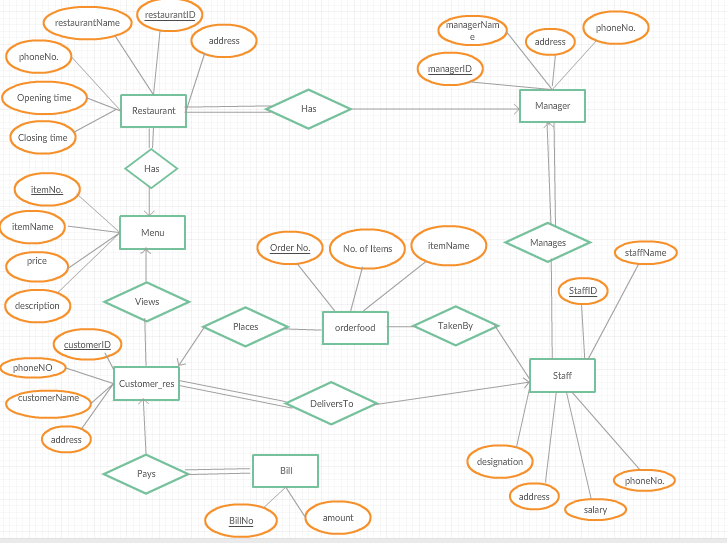
1. Restaurant - contains information specific to each restaurant
2. Menu - provides menu options and items on sale
3. Customer - holds demographic information about the customer
4. Bill - provides the bill number and amount
5. Order – holds order details such as; order number, number of items and item name
6. Staff – contains demographic information of the staff
7. Manager - contains demographic information about the manager
8. TakenBy – connects entities orderfood and staff

# Entity Relationship Design

## Requirements

* The Restaurant entity encompasses information for many different categories of restaurants.
* Each **restaurant** has its own restaurant\_ID, restaurant\_name, address, phone\_number, opening\_time, and closing\_time, manager\_ID, item\_number.
* The **manager** can have one or more restaurants and as well as he/she will be managing the staff. The manager entity will consist of manager\_ID, manager\_name, address, and phone\_number
* For each restaurant, there will be a designated **menu** that contains an item\_number that is unique for all orders. A menu can be viewed by one or many customers. Each customer will consist of item\_number, item\_name, description, and price.
* Next up is the **bill**, which the customer pays. Each bill consists of bill\_number, amount, and customer\_ID
* The customer would be placing orders which will be taken by the staff. The **orderfood** table contains order\_number, numberof\_item, item\_name, and customer\_ID
* A restaurant will have **staff** who will take orders given by the customers at the various restaurants. The staff will also be delivering the food to the customers and preparing the food. The staff entity encompasses of staff\_ID, staff\_name, designation, phone\_number, address, salary, and manager\_ID.
* The **customer\_res** table will include demographic details about the customers. It will contain the following attributes; customer\_ID, customer\_name, address, phone\_number, item\_number, and staff\_ID.
* Lastly, the **takenby** table will connect the entities orderfood and staff. It has a many to many relationship with both the entities. It contains attributes order\_number, and staff\_ID.

## Entity Relationship Diagram



## Assumptions

1. Each restaurant must have a manager and a manager could have more restaurants
2. Each restaurant must have a menu and a menu can be viewed by many customers
3. A customer can place many order and order can be placed by a customer
4. Order can be taken many staffs
5. A staff can deliver food to the many customers and customer must have a staff that delivers food to them
6. A customer can pay many bills and bills must be paid by a customer
7. A Manager must manage many staffs and staffs can be managed by a manager

# Schema and SQL Implementation

## Relational Schema

1. **Restaurant**

**restaurant (*restaurant\_ID:* INTEGER, *managerID:* INTEGER, *restaurant\_name*: VARCHAR (25), *address*: VARCHAR (255), *phone\_number*: VARCHAR (12), *itemNo:* VARCHAR(12), *opening\_time*: VARCHAR (10), closing\_time: VARCHAR (10))**

Restaurant table stores one entry for each restaurant. Each restaurant can be uniquely identified by a restaurant\_ID which is the primary key. Other information that it holds are; restaurant name, address, phone number, opening time and closing time of the restaurant.

1. **Customer**

**customer\_res (*customer\_ID:* INTEGER, *customer\_name:* VARCHAR (60), *address:* VARCHAR (255), *item\_number:* VARCHAR(12), *staff\_ID:* INTEGER, *phone\_number:* VARCHAR (12))**

Customer table stores one entry for each customer. Each customer can be identified by a unique customer\_ID which is the primary key. Customer table also stores other information about the customer, such as; customer name, address and phone number.

1. **Menu**

**menu (*item\_number:* INTEGER, *item\_name:* VARCHAR (35), *description:* VARCHAR (250), *price:* INTEGER))**

Menu table stores the information of food items that are served at the restaurant. Each menu has an item\_number which is the primary key of the entity. It also stores the item name, description of the food item and its price.

1. **Manager**

**manager (*manager\_ID*: INTEGER, *manager\_name*: VARCHAR (60), *address:* VARCHAR (255), *phone\_number*: VARCHAR (12), *restaurant\_ID*: INTEGER))**

Manager table stores one entry for each Manager. Each manager can be uniquely identified by manager\_ID which is the primary key. This table will also store a manager’s personal information such as name, address, and phone number and restaurant ID.

1. **Staff**

**staff (*staff\_ID:* INTEGER, *staff\_name:* VARCHAR (50), *designation:* VARCHAR (35), *phone\_number:* VARCHAR (12), *address:* VARCHAR (50), *salar*y: FLOAT (10, 2), *manager\_ID:* INTEGER))**

Staff table can store one entry for each employee. Each employee can be uniquely identified by staff ID which is the primary key. Other information that is stored in the staff table are; name, designation, phone number, address and salary. Also, manager ID acts as a foreign key here so that a staff is directly associated with a manger that he/she reports to.

1. **Orderfood**

**orderfood (*order\_number:* INTEGER, *numberof\_items:* INTEGER, *item\_name:* VARCHAR (250), *customer\_ID* INTEGER))**

Orderfood table stores one entry for each order. Every order can be uniquely identified by its order number which is the primary key. Other order details such as number of items, item name will be stored in the table to further provide granular details about an order. The customer ID which is a forerign key referencing to the “customer\_res” table will also be included in the order table so that it is effortless to associate an order with a customer ID without joining tables.

1. **Bill**

**bill (*bill\_number*: INTEGER, *amount:* INTEGER, *customer\_ID:* INTEGER))**

Bill table stores one entry for each bill paid by a customer. Each bill can be uniquely identified by its bill number which is the primary key. This table has the total amount of the bill and the customer ID to effortlessly track each bill and the amount paid by a customer without joining tables.

1. **TakenBy**

TakenBy (OrderNo.:INTEGER, StaffID: INTEGER)

The TakenBy relationship connects entities orderfood and staff. The primary key from orderfood, order\_number is being used as a foreign key and staff\_ID which is a primary key from the staff table is being as a foreign key in this relationship.

## Implementation of the Schema in MariaDB

use lg13db;

# Database Table Creation

# First drop any existing tables. Any errors are ignored.

DROP TABLE IF EXISTS restaurant;

DROP TABLE IF EXISTS customer\_res;

DROP TABLE IF EXISTS menu;

DROP TABLE IF EXISTS manager;

DROP TABLE IF EXISTS staff,

DROP TABLE IF EXISTS orderfood;

DROP TABLE IF EXISTS bill;

# Now, add each table.

create table restaurant(

restaurant\_ID INTEGER,

managerID INTEGER,

restaurant\_name VARCHAR (25),

address VARCHAR (255),

phone\_number VARCHAR (12),

opening\_time VARCHAR (10),

closing\_time VARCHAR (10),

item\_number INTEGER,

Primary Key(restaurant\_ID)

Foreign Key(managerID) references manager (managerID)

Foreign Key(item\_number) references menu (item\_number)

ON DELETE NO ACTION

ON UPDATE NO ACTION

)ENGINE=InnoDB;

# If updates or deletes happen to the foreign key, there will be no action

create table customer\_res(

customer\_ID INTEGER,

customer\_name VARCHAR(60),

address VARCHAR(255),

phone\_number VARCHAR(12),

staff\_ID INTEGER,

item\_number INTEGER,

Primary Key(customer\_ID)

Foreign Key (staff\_ID) references staff (staff\_ID)

Foreign Key(item\_number) references menu (item\_number)

ON DELETE NO ACTION

ON UPDATE NO ACTION

)ENGINE=InnoDB;

# If updates or deletes happen to the foreign key, there will be no action

create table menu(

item\_number INTEGER,

item\_name VARCHAR(35),

description VARCHAR(250),

price INTEGER,

Primary Key(item\_number)

)ENGINE=InnoDB;

create table manager(

manager\_ID INTEGER,

manager\_name VARCHAR(60),

address VARCHAR(255),

phone\_number VARCHAR(12),

Primary Key(manager\_ID),

)ENGINE=InnoDB;

create table takenby(

order\_number INTEGER,

staff\_ID INTEGER,

Foreign Key(order\_number) references orderfood(order\_number),

Foreign Key(staff\_ID) references staff(staff\_ID)

)ENGINE=InnoDB;

create table staff(

staff\_ID INTEGER,

staff\_name VARCHAR(50),

designation VARCHAR(35) NOT NULL,

phone\_number VARCHAR(12),

address VARCHAR(50),

salary FLOAT(10, 2),

manager\_ID INTEGER,

Primary Key(staff\_ID),

Foreign Key(manager\_ID) references manager(manager\_ID)

ON DELETE NO ACTION

ON UPDATE NO ACTION

)ENGINE=InnoDB;

# If updates or deletes happen to the foreign key, there will be no action

create table orderfood(

order\_number INTEGER,

numberof\_items INTEGER,

item\_name VARCHAR(250),

customer\_ID INTEGER,

Primary Key(order\_number),

Foreign Key(customer\_ID) references customer\_res(customer\_ID)

ON DELETE NO ACTION

ON UPDATE NO ACTION

)ENGINE=InnoDB;

# If updates or deletes happen to the foreign key, there will be no action

create table bill(

bill\_number INTEGER,

amount INTEGER,

customer\_ID INTEGER NOT NULL,

Primary Key(bill\_number),

Foreign Key(customer\_ID) references customer\_res(customer\_ID)

ON DELETE NO ACTION

ON UPDATE NO ACTION

)ENGINE=InnoDB;

# If updates or deletes happen to the foreign key, there will be no action

**Data Files**

restaurant

001,Edo Japanese Restaurant and Bar,1 Residency Road Fairfax VA 560025,301-854-9654,11 AM,10 PM,029,049

002,Karavalli,8130 Wisconsin Ave Bethesda MD 20814,240-587-6323,11 AM,10 PM,028,048

003,Persian Terrace,7905 Norfolk Ave Bethesda MD 20814,301 215-7189,10 AM,11 PM,027,047

004,JW Kitchen,7239 Woodmont Ave Bethesda VA 20814,240-555-1010,10 AM,9 PM,026,046

005,Pasta Street,4901 B Fairmont Ave Bethesda MD 20814,301-878-0233,11 AM,10 PM,025,045

006,Alto Vino,7814 Old Georgetown Rd Falls Church VA 20814,240-969-3325,11 AM,10 PM,024,044

007,Blue Ginger,7501 Wisconsin Ave Bethesda MD 20815,301-414-8745,10 AM,11 AM,023,043

008,The Lantern,4867 Cordell Ave Bethesda VA 20814,301-857-9898,10 AM,11 PM,022,042

009,Brick Oven,4910 Elm St Bethesda MD 20814,301-323-2121,9 AM,10 PM,021,041

010,Bombay Cafe,4829 Bethesda Ave Bethesda VA 20814,240-727-1247,9 AM,10 PM,020,040

customer\_res

049,Mike Tyson,3222 Fort Street Rocky Mount NC 33043,240-878-6954,049,100

050,Patricia Kilmer,3222 Fort Street Rocky Mount NC 33043,240-878-6954,049,099

051,Virginia Blair,509 Logan Lane Denver CO 80265,301-777-2358,048,098

052,Jose Prado,765 Marshall Street Baltimore MD 21231,301-999-2024,047,097

053,Joseph B Hill,1673 Larry Street Milwaukee WI 53210,454-897-9999,046,096

054,Marie Cooper,1473 Lyndon Street Tremont PA 17981,874-978-4777,045,095

055,Brenda I Hebert,1468 Swick Hill Street Concord NC 28025,301-800-9898,044,094

056,Irene Roberts,1379 Broadcast Drive Chantilly VA 22021,240-874-6363,043,093

057,Fabian N Alfonso,783 Davis Court Stlouis IL 63101,301-888-5454,042,092

058,Robert Renner,475 Eagle Drive Southfield MI 48075,240-765-9812,041,091

059,Janet P Potts,2355 Patterson Street Houston TX 77002,301-633-8973,040,095

menu

040,Avocado Salad,Avocado with Special Sauce,17

041,Sushi Appetizer,Tuna Salmon Shrimp White Fish,28

042,Sushi lunch combo,Assorted Fish with Chef Special hot sauce,10

043,Tempura Appetizer,Shrimp and Assorted Veggies,8

044,Spring roll,Shrimp Tempura Eel Cucumber egg with Cream Cheese,19

045,Chicken Teriyaki,Charcoal-broiled boneless Chicken served in Teriyaki sauce,12

046,Beef Teriyaki,Thin-sliced Beef marinated and cooked in our special sauce,15

047,Teriyaki Beef Ribs,3 sliced Ribs marinated and served charcoal broiled,13

048,Happy Bowl,Vegetables and noodles in Sukiyaki sauce with Beef over Rice,11

049,Spicy Vegetable with Chicken,Mixed Vegetable stir-fried with Chicken,10

manager

020,Stever Brunce,29 Maryland Ave Rockville MD 20850,301-742-3782

021,Keith Hill, 12 East Diamond Avenue Gaithersburg MD 20877,240-878-2525

022,Mark Robins,9021 Gaither Rd Gaithersburg MD 20877,240-254-9688

023,Danny Wilson,7421 Maple Lawn Blvd Fulton MD 20759,301-969-3454

024,Neal Kumar, 228 N Market St Frederick MD 21701,301-636-7423

025,Lee Clark,18224 Village Center Dr Olney MD 20832,240-722-6963

026,Gary Johnson,12907 Wisteria Dr Germantown MD 20874,240-999-0202

027,Dave Jones,3308 Olney Sandy Spring Rd Olney MD 20832,301-666-8454

028,Phil Parkinson,18530 Woodfield Rd Gaithersburg MD 20879,301-974-3344

029,Bob Peters, 7618 Main St Sykesville MD 21784,240-452-8977

staff

091,Angela R Washington,Head Chef,803-245-4948,2831 Marion Street Bamberg SC 29003,60000,020

092,Brenda F Grubb,Chef,706-719-5689,4498 Radio Park Drive Athens GA 30601,50000,021

093,Harold Mead,Server,972-646-8586,362 Worthington Drive Bardwell TX 75165,40000,022

094,Bradley Jett,Server,856-290-5266,1148 Lee Avenue Camden NJ 08102,40000,023

095,Earl Grider,Server,239-860-8299,2070 Village View Drive Fort Myers FL 33912,40000,024

096,Kevin L Steere,Bus person,410-793-7114,587 Woodhill Avenue Crofton MD 21114,35000,025

097,Tyler A Wright,Bus person,407-983-1561,3127 Hide A Way Road Winter Park FL 32789,35000,026

098,Wilson M Vargas,Host,405-752-9529,436 Ruckman Road Oklahoma City OK 73113,35000,027

099,Wanda C Armstrong,Cashier,617-376-2463,1798 Aspen Court Quincy MA 02169,35000,028

100,Emma Ling,Bartender,603-318-1646,756 Shearwood Forest Drive Bedford NH 01730,40000,029

orderfood

070,2,Happy Bowl,050

071,1,Spicy Vegetable with Chicken,051

072,2,Spicy Vegetable with Chicken,052

073,1,Chicken Teriyaki,053

074,1,Beef Teriyaki,054

075,1,Sushi Appetizer,055

076,3,Avocado Salad,056

077,1,Beef Teriyaki,057

078,1,Chicken Teriyaki,058

079,1,Teriyaki Beef Ribs,059

bill

069,20,049

068,30,050

067,35,051

066,50,052

065,55,053

064,60,054

063,25,055

062,35,056

061,15,057

060,10,058

takenby

070,100

071,099

072,098

073,097

074,096

075,095

076,094

077,093

078,092

079,091

User Interface Design

May 10, 2018

## Create User Interface Design:

A screenshot of a cell phone screen with text

Description generated with high confidence

## Main Functionality of Each Page:

Management Portal Tab:

This is the main management tab and is more related to management usage. This tab contains six subpages. The detail description and function of each sub page is described below.

* Delete Restaurant Information Page:

For each restaurant listed in our data base, management team would be able to delete any information related to Restaurant by using unique restaurant ID and manager ID.

* Add Staff Information Page:

Management team would be able to add new staff information in the database such as staff name, staff ID, salary, designation etc., When the restaurant hires new staff, the management team could add staff in the system by providing all the required information.

* Delete Staff Information Page:

Management team has ability to delete entire staff profile from the database by providing unique staff ID associated with staff information.

* Update Staff Information Page:

The certain information related to staff can be updated in the database by management team by providing unique staff ID.

* Update Menu Page:

The menu page is used to add a new item to a menu by providing all required information.

* View Order Page:

For each order in our database, management team and staff would be able to see all the order.

* Staff Salary Information Page:

The management team would be able to look at the who salary is highest (in our case salary>=50,000).

Customer Portal Tab:

This tab includes all the informative pages that customers (users) can use to access our website. Customer would be able to view restaurant information, menu, make an order etc.,

* Restaurant Information Page:

For each restaurant in our database, users will be presented with information such as the restaurant name, address, opening time, closing time, and phone number.

* View Menu Page:

The customer will be able to see menu before making any decision for order.

* Make an Order Page:

By using the make an order page, customer will be able to place an order online by providing all the information. Every customer, who is using this system, should already have unique customer ID.

* Items on Sale Page:

The customer would be able to view all the items which are on sale. In our case we put the items whose price is less or equal to 15 dollars.

* View Order Page:

Customer would utilize this page to look at their order that they placed online.

About Us Page:

This page is to introduce and display our restaurant and restaurant management system.

Contact Us Page:

This page is to provide the contact information of restaurant and the related persons.

**3.** **Website Like to Our Homepage:**

<http://pluto.hood.edu/~lg13/index.html>