**CHAPTER 1**

**Description**

* The project title “Restaurant management database system” is designed SQL sever.
* There will be different items available in a restaurant ; many customers will be coming at different time for having food, they will be eating different items available in the restaurant.
* Restaurant owner has to maintain records of each sales.
* This system will save time and will be easy to use when compares to manual work will be done in papers.
* Restaurant management database system used by restaurant personal to collect data, process it and store it for future use.

**Objective**

* Excellent customers and personal service.
* To ensure the customers satisfaction and build a repeat customer base.
* To keep records of menu items, price details, suppliers details and others.
* The primary objective of the project is to develop a database on restaurant.

**CHAPTER-2**

**Project planning and scheduling**

**2.1 Project planning**

Effective management of a software project depends on thoroughly planning the progress of the project. The success criteria for project managements

obviously vary from project but for most projects, important goals are:

* deliver the product to the customer at the agreed time
* keep overall costs within the budget plan
* deliver product that meets the customer’s exception
* maintain the product quality and services for customer’s satisfying

**2.2 Project constraint**

* Project must be delivered on time.
* Project must meet the agreed scope – no more, no less.
* Project must keep required quality.

**2.3 Requirements**

**Software**

Window OS-XP/7/8/8.1/10

MySql

Gantt Chart Excel

**Hardware**

Processor : Petium or Higher

RAM : 1GB or Higher

**2.4 Risk management**

* Member turnover : Experienced team member will leave the project before it is finished.
* Hardware unavailability : Hardware which is essential for the project will not be delivered on schedule.
* Requirements change : There will be a larger number of changes to the requirements than anticipated.

**2.5 Project scheduling**

* Project scheduling involves separating the total work involved in a project into separate activities and judging the time required to complete these

activities.

* Some of these activities are carried out in parallel.

**2.5.1 Work Break Down Structure**

* The WBS is the tool that is used to record and communicate the project deliverables ( something produced or an outcome ) and sub-deliverables as well as the accomplishments and accomplishments.
* The identification of these elements relies on the experience of the team members as well as the consultation with outside experts.

**Work Break Down Structure for Restaurant Management System**

1. Create database designed

1.1 Draw ER diagram

1.1.1 Draw relational schema

1.2 Create data dictionary

2. Create database tables

2.1 Collect data for tables

2.2 Insert data into the tables

3. Create security constraints

3.1 Query the database

4. Create trigger

Figure – 2.1 : WBS for Restaurant Management System

**Work Break Down Structure (WBS)**

Restaurant Management System

2.

Create database tables

1.1

Draw ER diagram

4.

Create trigger

3.

Create security constraints

1.

Create database designed

2.2

Insertdata into the tables

2.1

Collect data for tables

1.2

Create data dictionary

1.1.1

Draw relational schema

3.1

Query the database

Figure – 2.2 : WBS for Restaurant Management System

**2.5.2 Task duration and dependencies**

T1= 1.Create database designed T6= 2.1 Collect data for tables

T2= 1.1Draw ER diagram T7= 2.2 Insert data into the tables

T3= 1.1.1 Draw relational schema T8= 3. Create security constraints

T4= 1.2 Create data dictionary T9= 3.1 Query the database

T5= 2. Create database tables T10= 4 Create trigger

|  |  |  |
| --- | --- | --- |
| **Task** | **Duration** | **Dependencies** |
| T1 | 1 |  |
| T2 | 1 | T1(M1) |
| T3 | 1 | T2(M2) |
| T4 | 1 | T1(M1) |
| T5 | 1 | T1(M1) |
| T6 | 2 | T5(M3) |
| T7 | 2 | T6(M4) |
| T8 | 1 | T7(M5) |
| T9 | 2 | T8(M6) |
| T10 | 3 | T8(M6) |

Figure – 2.3 : Task durations and dependencies

**2.5.2.1 Determination of the Critical Path**

1days 1 days

M1

M3

T5

T1

T2

T3

T4

T6

T7

T8

T9

T10

M2

M4

M5

M6

Stop

1/2/2019 1days 1days 2days 2 days 1 days

Start

1 days 2 days 3 days

Figure – 2.4 : Critical path diagram for Restaurant Management System

The minimum time required to finish the Restaurant Management System project is 10 days.

**2.5.2.2 Create the schedule**

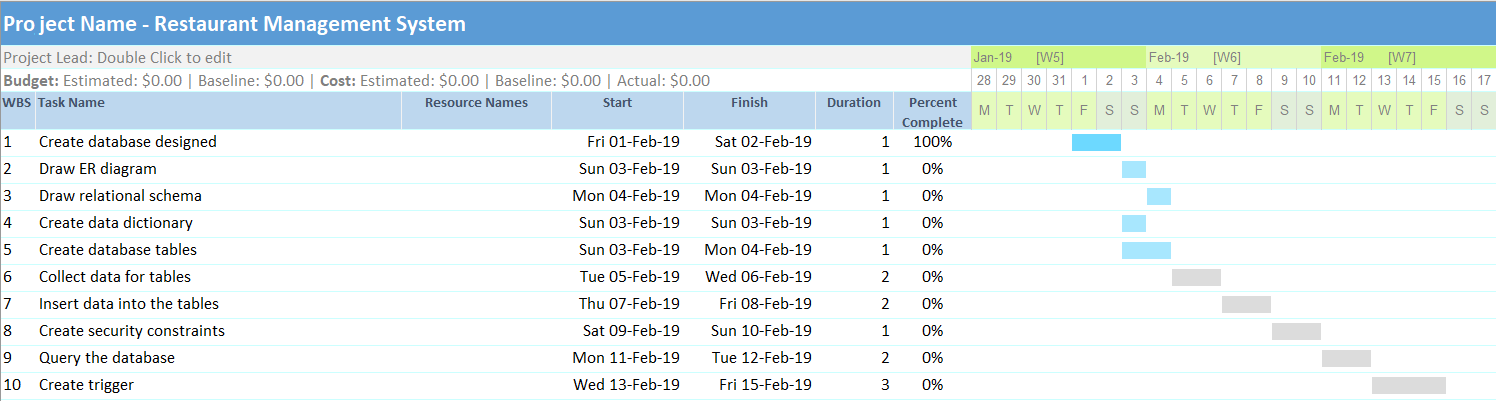


Figure – 2.5 : Schedule and Gantt Chart for Restaurant Management System

**CHAPTER-3**

**Create data & Insert data**

**3.1 Creating database**

mysql > create database restaurant;

mysql > use restaurant;

**3.2 Creating table**

Creating for Customer table

Create table Customer(Cid varchar(5),Name varchar(30),Address varchar(50),Contact varchar(15));

Creating for Chief\_prepare table

Create table Chief\_prepare(Sid varchar(5),Ono varchar(5));

Creating for Staff table

Create table Staff(Sid varchar(5),Name varchar(30),Address varchar(50),Contact varchar(15),Age int,Position int,Salary int);

Creating for Waiter\_Deliver table

Create table Waiter\_Deliver(Sid varchar(5),Cid varchar(5),Ono varchar(5));

Creating for OrderDetail table

Create table OrderDetail(Ono varchar(5),Fid varchar(5),Cid varchar(5),Quantity int,Amount int,Order\_date varchar(10));

Creating for Bill table

Create table Bill(Bid varchar(5),Cid varchar(5),Ono varchar(5),Totalprice int);

Creating for Food table

Create table Food(Fid varchar(5),Name varchar(30),Type varchar(50),Price int);

**3.3 Inserting data**

**Customer**

|  |  |  |  |
| --- | --- | --- | --- |
| Cid | Name | Address | Contact |
| C1 | Khin Nyein Chan | Myoma | 09-2234453 |
| C2 | Ay Sea | Mine Khat | 09-7865434 |
| C3 | Ji Aung | 2 Mile | 09-2543563 |
| C4 | Nway Oo San | Manpher | 09-6754445 |
| C5 | Zin Thu Myint | Hinthar | 09-7865434 |
| C6 | Thandar Than | Si In | 09-7865455 |
| C7 | Khaing linn wai | Moe Mouk | 09-7865486 |
| C8 | Khin thuzar win | Si In | 09-7865474 |
| C9 | Ayme tun | Si In | 09-7865494 |
| C10 | Tin Tin Shwe | Si In | 09-7865444 |

**Chief\_prepare**

|  |  |
| --- | --- |
| Sid | Ono |
| S10 | 0001 |
| S11 | 0002 |
| S12 | 0005 |
| S13 | 0009 |
| S11 | 0003 |
| S13 | 0007 |
| S10 | 0006 |
| S12 | 0004 |
| S10 | 0008 |

**Staff**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sid | Name | Address | Contact | Age | Position | Salary |
| S1 | U Ko | No-24,Rose Street, Banmaw | 09-110011 | 37 | Admin | 0 |
| S2 | U Hla | No-5,Thiri Street, Banmaw | 09-124567 | 30 | Chef | 2000000 |
| S3 | U Tun | No-10,Thukha Street, Banmaw | 09-256017 | 33 | Chef | 2000000 |
| S4 | U Khant | No-16,Main Street, Banmaw | 09-75413 | 30 | Chef | 2000000 |
| S5 | U Zaw | No-4,Padauk Street, Banmaw | 09-89496 | 32 | Chef | 2000000 |
| S6 | U Ba | No-12,Orchid  Street, Banmaw | 09-120456 | 35 | Manager | 300000 |
| S7 | Mg Mg | No-3,Jasmine Street, Banmaw | 09-25645 | 20 | Waiter | 70000 |
| S8 | Ma Ma | No-10,Rose Street, Banmaw | 09-25314 | 22 | Waiter | 70000 |
| S9 | Ko Ko | No-1,Padauk Street, Banmaw | 09-24137 | 25 | Waiter | 70000 |
| S10 | Kyaw Kyaw | No-5,Orchid Street, Banmaw | 09-76245 | 19 | Waiter | 70000 |
| S11 | Su Su | No-50,Jasmine  Street, Banmaw | 09-42645 | 21 | Waiter | 70000 |
| S12 | Nilar | No-42,Main Street, Banmaw | 09-24017 | 20 | Waiter | 70000 |
| S13 | U Aye | No-13,Poppy Street, Banmaw | 09-20145 | 28 | Chef | 2000000 |

**Waiter\_Deliver**

|  |  |  |
| --- | --- | --- |
| Sid | Cid | Ono |
| S1 | C1 | 0001 |
| S2 | C2 | 0002 |
| S3 | C6 | 0004 |
| S4 | C5 | 0005 |
| S5 | C7 | 0006 |
| S6 | C4 | 0007 |
| S7 | C9 | 0008 |
| S8 | C10 | 0009 |
| S9 | C8 | 0003 |
| S10 | C2 | 0002 |

**OrderDetail**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ono | Fid | Cid | Quantity | Amount | Order\_date |
| 0001 | F1 | C1 | 2 | 3000 | 1\_1\_2019 |
| 0001 | F2 | C1 | 4 | 4000 | 1\_1\_2019 |
| 0002 | F2 | C2 | 5 | 5000 | 5\_1\_2019 |
| 0002 | F3 | C2 | 3 | 4500 | 5\_1\_2019 |
| 0002 | F2 | C2 | 2 | 7000 | 5\_1\_2019 |
| 0003 | F1 | C8 | 5 | 7500 | 10\_2\_2019 |
| 0004 | F3 | C6 | 3 | 4500 | 11\_3\_2019 |
| 0004 | F5 | C6 | 6 | 12000 | 11\_3\_2019 |
| 0005 | F7 | C5 | 5 | 15000 | 15\_3\_2019 |
| 0005 | F9 | C5 | 4 | 12000 | 15\_3\_2019 |
| 0006 | F8 | C7 | 5 | 7500 | 16\_2\_2019 |
| 0007 | F7 | C4 | 7 | 10500 | 18\_2\_2019 |
| 0007 | F10 | C4 | 10 | 30000 | 18\_2\_2019 |
| 0008 | F12 | C9 | 5 | 7500 | 20\_3\_2019 |
| 0009 | F13 | C10 | 4 | 4000 | 25\_3\_2019 |

**Bill**

|  |  |  |  |
| --- | --- | --- | --- |
| Bid | Cid | Ono | Totalprice |
| B1 | C1 | 0001 | 7000 |
| B2 | C2 | 0002 | 11500 |
| B3 | C4 | 0007 | 40500 |
| B4 | C5 | 0005 | 27000 |
| B5 | C6 | 0004 | 16500 |
| B6 | C7 | 0006 | 7500 |
| B7 | C8 | 0003 | 7500 |
| B8 | C9 | 0008 | 7500 |
| B9 | C10 | 0009 | 4000 |

**Food**

|  |  |  |  |
| --- | --- | --- | --- |
| Fid | Name | Type | Price |
| F1 | Fried Noodle | Food | 1500 |
| F1 | Donut | Drink | 3000 |
| F1 | Pizza | Food | 3000 |
| F1 | Fire Dragon | Drink | 1500 |
| F1 | Strawberry Juice | Drink | 1500 |
| F1 | Ice Cream | Food | 1500 |
| F1 | Coffee | Drink | 1000 |
| F1 | Lichee Juice | Drink | 1000 |
| F1 | Pudding | Food | 1500 |
| F1 | Hamburgar | Food | 2500 |
| F1 | Potato Chip | Food | 3000 |
| F1 | Sandwich | Food | 3000 |
| F1 | Lemonade | Drink | 1500 |
| F1 | Coke | Drink | 1500 |
| F1 | Milkshake | Drink | 3000 |

**CHAPTET-4**

**DATABASE DESIGN**

**4.1 Data dictionary**

**Chief\_prepare**

|  |  |  |  |
| --- | --- | --- | --- |
| SID | Type | PK/FK | Null |
| Sid | Varchar(5) |  | Yes |
| Ono | Varchar(5) |  | Yes |

**Customer**

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Type | PK/FK | Null |
| Cid | Varchar(5) | PK | Yes |
| Name | Varchar(30) |  | Yes |
| Address | Varchar(50) |  | Yes |
| Contact | Varchar(15) |  | Yes |

**Food**

|  |  |  |  |
| --- | --- | --- | --- |
| FID | Type | PK/FK | Null |
| Fid | Varchar(5) | PK | Yes |
| Name | Varchar(30) |  | Yes |
| Type | Varchar(50) |  | Yes |
| Price | Int(11) |  | Yes |

**Staff**

|  |  |  |  |
| --- | --- | --- | --- |
| SID | Type | PK/FK | Null |
| Sid | Varchar(5) | PK | Yes |
| Name | Varchar(30) |  | Yes |
| Address | Varchar(50) |  | Yes |
| Contact | Varchar(15) |  | Yes |
| Age | Int(11) |  | Yes |
| Position | Int(11) |  | Yes |
| Salary | Int(11) |  | No |

**OrderDetail**

|  |  |  |  |
| --- | --- | --- | --- |
| ONO | Type | PK/FK | Null |
| Ono | Varchar(5) |  | Yes |
| Fid | Varchar(5) | FK | Yes |
| Cid | Varchar(5) | FK | Yes |
| Quantity | Int(11) |  | Yes |
| Amount | Int(11) |  | Yes |
| Order-date | Varchar(10) |  | Yes |

**Bill**

|  |  |  |  |
| --- | --- | --- | --- |
| BID | Type | PK/FK | Null |
| Bid | Varchar(5) | PK | Yes |
| Cid | Varchar(5) | FK | Yes |
| Ono | Varchar(5) |  | Yes |
| TotalPrice | Int(11) |  | Yes |

**Waiter\_Deliver**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Type | PK/FK | Null |
| Sid | Varchar(5) | FK | Yes |
| Fid | Varchar(5) | FK | Yes |
| Ono | Varchar(5) |  | Yes |

**4.2 ER Diagram**

Contain

Containn

Customer

Staff

Chief\_prepare

Waiter\_Deliver

Prepare

Taken By

By

Deliver

OrderDetails

Contain

Order

Pay

Bill

Food

Bill

Bid

Cid

Ono

TotalPrice

Food

Fid

Name

Type

Price

OrderDetail

Ono

Fid

Cid

Quantity

Amount

Order\_date

Waiter\_Deliver

Sid

Cid

Ono

Staff

Sid

Name

Address

Contact

Age

Position

Salary

Customer

Cid

Name

Address

Contact

Chief\_prepare

Sid

Ono

**1…\* 1..\* 1..\***

**1..\* 1..\* 1..\* 1..\* 1..\* 1..\* 1..\* 1..\* 1..\* 1..\* 1..\* 1..\*1..\* 1..\* 1..\* 1..\***

**1..\* 1..\***

**Relational schema**

Create table Customer(Cid varchar(5),Name varchar(30),Address varchar(50),Contact varchar(15));

Create table Chief\_prepare(Sid varchar(5),Ono varchar(5));

Create table Staff(Sid varchar(5),Name varchar(30),Address varchar(50),Contact varchar(15),Age int,Position int,Salary int);

Create table Waiter\_Deliver(Sid varchar(5),Cid varchar(5),Ono varchar(5));

Create table OrderDetail(Ono varchar(5),Fid varchar(5),Cid varchar(5),Quantity int,Amount int,Order\_date varchar(10));

Create table Bill(Bid varchar(5),Cid varchar(5),Ono varchar(5),Totalprice int);

Create table Food(Fid varchar(5),Name varchar(30),Type varchar(50),Price int);

**4.3 Data Flow Diagram**

Customer view foods

Customer select foods

Order?

Start

Record order detail

Chief prepare ordered food

Waiter delivered food to customer

Database

More order?

Customer pay bill

Stop

no yes

yes

**CHAPTER 5**

**Trigger and Security**

**5.1 Identify the Users and Passwords**

Creating user for admin

<mysql> create user manager@localhost identified by'knc';

Query OK, 0 rows affected (0.25 sec)

Creating user for customer

<mysql> create user customer@localhost identified by 'cust';

Query OK, 0 rows affected (0.00 sec)

**5.2 Define view and privileges**

<mysql> grant select,insert,update,delete on restaurant.\* to manager@localhost identified by 'knc' with grant option;

Query OK, 0 rows affected (0.00 sec)

<mysql> grant select on Food to customer@localhost identified by 'cust';

Query OK, 0 rows affected (0.01 sec)

**Using cmd client**

**For manager**

Microsoft Windows [Version 10.0.17134.590]

(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\DELL>cd/

C:\>cd..

C:\>"C:\Program Files\MySQL\MySQL Server 5.5\bin\mysql.exe" -u manager -p

Enter password: \*\*\*

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 10

Server version: 5.5.16 MySQL Community Server (GPL)

Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its

affiliates. Other names may be trademarks of their respective

owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

<mysql> use restaurant;

Database changed

**1.Get all food information.**

<mysql> select \* from Food;

QueryOK,1rowaffected(0.13sec) ****

**2.Delete from food where name is ‘Hot Dog’.**

<mysql> delete from Food where Name='Hot Dog';

Query OK, 1 row affected (0.13 sec)



**3.Update contact from staff where Name=’U Zaw’ to ’09-89496’.**

<mysql> update Staff set Contact='0989496' where Name='U Zaw';

Query OK, 1 row affected (0.07 sec)



**4.Insert new food with values(F16,Coffee,Drink,1000).**

<mysql> insert into Food values('F16','Coffee','Drink',1000);

Query OK, 1 row affected (0.05 sec)



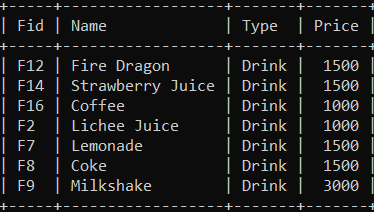
**For Customer**

**1.Select all information of items in food which type are ‘Drink’.**

<mysql> select \* from Food where Type='Drink';

Query OK, 1 row affected (0.06 sec)

7 rows in set (0.00 sec)



**2.Change the price from food where Fid=’F16’ to 2000.**

<mysql> update Food set Price=2000 where Fid='F16';

ERROR 1142 (42000): UPDATE command denied to user 'customer'@'localhost' for table 'food'

**3.Add a new item in food (‘F17’,’Green Soda’,’Drink’,2000).**

<mysql> insert into Food values('F17','Green Soda','Drink',2000);

ERROR 1142 (42000): INSERT command denied to user 'customer'@'localhost' for table 'food'

**4.Delete from food where Name=’Fire Dragon’ and Price=1500.**

<mysql> delete from Food where Name='Fire Dragon' and Price=1500;

ERROR 1142 (42000): INSERT command denied to user 'customer'@'localhost' for table 'food'

**5.Get all staff information.**

<mysql> select \* from Staff;

ERROR 1142 (42000): SELECT command denied to user 'customer'@'localhost' for table 'staff'

**5.3 Create trigger for update**

**For Food Table**

**1.Create another table “Food\_Info” with field name and data type respectively. Food\_Info(fid int,name varchar(30),type varchar(20),old\_price int,new\_price int,Timestamp default current\_timestamp)**

<mysql> create table Food\_Info(Fid varchar(5),Name varchar(30),Type varchar(20),Old\_price int,New\_price int,Data\_Modified timestamp default current\_timestamp);

Query OK, 0 rows affected (0.10 sec)

**2.The create Trigger create ‘Food\_Trigger’ on table ‘Food’.The Trigger ‘Food\_Trigger’ field automatically BEFORE UPDATE operation is performed on table ‘Food’. Futhermore, the update record is inserted into table Food\_Info.**

<mysql> create trigger Food\_Trigger before update on Food for each row insert into Food\_Info(Fid,Name,Type,Old\_price,New\_price) values(old.Fid,old.Name,old.Type,old.Price,new.Price);

Query OK, 0 rows affected (0.12 sec)

**5.4 Updating data**

**3.Change number of price 2500 where food name is “Hamburgar”.**

<mysql> update Food set Price=2500 where Name='Hamburgar';

Query OK, 1 row affected (0.10 sec)

Rows matched: 1 Changed: 1 Warnings: 0

**4.Get query to view Food\_Info table.**

<mysql> select \* from Food\_Info;

+------+-----------+------+-----------+-----------+---------------------+

| Fid | Name | Type | Old\_price | New\_price | Data\_Modified |

+------+-----------+------+-----------+-----------+---------------------+

| F4 | Hamburgar | Food | 2000 | 2500 | 2019-03-07 21:40:35 |

**+------+-----------+------+-----------+-----------+---------------------+**

3 rows in set (0.03 sec)

**5.Write drop trigger for Food\_Info.**

<mysql>drop trigger Food\_Trigger;

Query OK, 0 rows affected (0.01 sec)