

ITAHARI
INTERNATIONAL
COLLEGE

Module Code & Module Title

CC6001NT Advanced Database System Development

Assessment Weightage & Type

40% Individual Coursework

Year and Semester

2020 Autumn

Student Name: Ngamesh Raj Bhandari

London Met ID: 18028912

College ID: NP05CP4A180036

Assignment Due Date: 19th March 2020

Assignment Submission Date: 19th March 2020

Academic Supervisor: Mr. Lekhnath Katuwal

Word Count (Where Required):

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Acknowledgment

It has been a great pleasure doing this coursework. I would like to express my special thanks of gratitude to my module teacher Mr. Lekhnath Katwal Sir who enlightened us with the glamorous knowledge of advance database. I am much grateful being his student. I would also like to thank my classmates who motivated and encouraged me to do my best.

Last but not the least, I would also like to thank our college Itahari International for providing us with coursework like this which helps us build inside and out.

Finally, a special thanks goes to my parents for the constant love and support I received throughout this project and friends who helped me a lot in finalizing this project within the limited time frame.

Contents

1. Introduction	1
2. Textual Analysis and Preparation of Initial ERD	2
3. Normalization	4
3.1 Normalization of Fig 1.....	4
3.2 Normalization of Fig 2.....	7
4. Assumptions and Integration.....	10
5. Overall Entities	11
6. ER-Model	14
7. Script	15
8. Script Execution	20
9. Data Dictionary.....	29
10. INSERT Statement.....	38
11. Commit.....	47
12. SELECT Statement.....	48
13. User Manual.....	55
14. Testing	57
Testing 1: Editing values in Dish form.	57
Testing 2: Deleting values in dish form.....	59

Testing 3: Inserting values in dish form.....	60
Testing 4: Editing values in restaurant form.....	62
Testing 5: Deleting in restaurant.....	64
Testing 6: Inserting in restaurant	65
Testing 7: Edit in customer form.....	67
Testing 8: Deleting in customer form.....	69
Testing 9: Inserting in customer form.....	70
Testing 10: Editing values in loyalty points form.....	72
Testing 11: Deleting in loyalty points form.....	74
Testing 12: Inserting values in loyalty points form.....	76
Testing 13: Editing values in Delivery address form.....	78
Testing 14: Deleting in Delivery Address form.	80
Testing 15: Inserting in Delivery address form.	81
Testing 16: Data filtering on customer order details form.....	84
Testing 17: Data filtering on dish search form.	85
Testing 18: Data filtering on Order Activity form.....	86
Testing 19: Foreign Key violation.	87
Testing 20: Format violatixon.	89
15. Further Discussion	91
References	92

Figure 1 Individual entity relationship.....	2
Figure 2 Complete initial ERD.....	3
Figure 3 Final ERD.....	14
Figure 4 DDL script of Loyalty_points generated from DATA Modeler.....	15
Figure 5 DDL script of dish generated from DATA Modeler.....	15
Figure 6 DDL script of delivery_address generated from DATA Modeler.....	16
Figure 7 DDL script of order_register generated from DATA Modeler.....	16
Figure 8 DDL script of restaurant generated from DATA Modeler.....	17
Figure 9 DDL script of customer generated from DATA Modeler.....	17
Figure 10 DDL script of restaurent_dish generated from DATA Modeler.....	18
Figure 11 DDL script of order generated from DATA Modeler.....	18
Figure 12 DDL script of dish_order generated from DATA Modeler.....	19
Figure 13 Execution of DDL generated script of loyalty_points	20
Figure 14 DDL generated script of loyalty_points is executed, and table is created successfully.....	20
Figure 15 Execution of DDL generated script of delivery_address	21
Figure 16 DDL generated script of delivery_address is executed, and table is created successfully.....	21
Figure 17 Execution of DDL generated script of order_register	22
Figure 18 DDL generated script of order_register is executed, and table is created successfully.....	22

Figure 19 Execution of DDL generated script of restaurant	23
Figure 20 DDL generated script of restaurant is executed, and table is created successfully.....	23
Figure 21 Execution of DDL generated script of dish	24
Figure 22 DDL generated script of dish is executed, and table is created successfully.	24
Figure 23 Execution of DDL generated script of customer	25
Figure 24 DDL generated script of customer is executed, and table is created successfully.....	25
Figure 25 Execution of DDL generated script of restaurant_dish	26
Figure 26 DDL generated script of restaurant_dish is executed, and table is created successfully.....	26
Figure 27 Execution of DDL generated script of order	27
Figure 28 DDL generated script of order is executed, and table is created successfully.	27
Figure 29 Execution of DDL generated script of dish_order	28
Figure 30 DDL generated script of dish_order is executed, and table is created successfully.....	28
Figure 31 Insertion of statements in Loyalty_points table.....	38
Figure 32 Data are inserted in loyalty_points table successfully.	38
Figure 33 Insertion of statements in Dish table.	39
Figure 34 Data are inserted in Dish table successfully.....	39
Figure 35 Insertion of statements in Restaurant table.	40

Figure 36 Data are inserted in loyal table successfully.....	40
Figure 37 Insertion of statements in Delivery_address table.....	41
Figure 38 Data are inserted in Delivery_address table successfully.....	41
Figure 39 Insertion of statements in Order_register table.....	42
Figure 40 Data are inserted in Order_register table successfully.....	42
Figure 41 Insertion of statements in Customer table.....	43
Figure 42 Data are inserted in Customer table successfully.....	43
Figure 43 Insertion of statements in Restaurant_dish table.....	44
Figure 44 Data are inserted in loyal Restaurant_dish successfully.....	44
Figure 45 Insertion of statements in Order table.....	45
Figure 46 Data are inserted in Order table successfully.....	45
Figure 47 Insertion of statements in Dish_order table.....	46
Figure 48 Data are inserted in Dish_order table successfully.....	46
Figure 49 Commit is completed successfully.....	47
Figure 50 All data of loyalty_points table is displayed.....	48
Figure 51 All data of dish table is displayed.....	49
Figure 52 All data of restaurant table is displayed.....	49
Figure 53 All data of delivery_address table is displayed.....	50
Figure 54 All data of order_register table is displayed.....	50
Figure 55 All data of customer table is displayed.....	51
Figure 56 All data of restaurant_dish table is displayed.....	52
Figure 57 All data of order table is displayed.....	53
Figure 58 All data of dish_order table is displayed.....	54

Figure 59 User Guide	55
Figure 60 Before editing highlighted values in Dish form.	57
Figure 61 while editing highlighted values in Dish form.	58
Figure 62 After editing highlighted values in Dish form.	58
Figure 63 Before deleting highlighted values in Dish form.	59
Figure 64 After deleting highlighted values in Dish form.	59
Figure 65 Before inserting values in Dish form.	60
Figure 66 While inserting highlighted values in Dish form.	60
Figure 67 After inserting highlighted values in Dish form.	61
Figure 68 Before editing highlighted values in restaurant form.	62
Figure 69 While editing highlighted values in restaurant form.	62
Figure 70 After editing highlighted values in restaurant form.	63
Figure 71 Before deleting highlighted values in restaurant form.	64
Figure 72 After deleting highlighted values in restaurant form.	64
Figure 73 Before inserting values in restaurant form.	65
Figure 74 While inserting highlighted values in restaurant form.	65
Figure 75 After inserting highlighted values in restaurant form.	66
Figure 76 Before editing highlighted values in customer form.	67
Figure 77 While editing highlighted values in customer form.	67
Figure 78 After editing highlighted values in customer form.	68
Figure 79 Before deleting highlighted values in customer form.	69
Figure 80 After deleting highlighted values in customer form.	69
Figure 81 Before inserting highlighted values in customer form.	70

Figure 82 while inserting highlighted values in customer form.....	70
Figure 83 After inserting highlighted values in customer form.	71
Figure 84 Before editing highlighted values in loyalty points form.	72
Figure 85 While editing highlighted values in loyalty points form.....	72
Figure 86 After editing highlighted values in loyalty points form.	73
Figure 87 Before deleting highlighted values in loyalty points form.	74
Figure 88 After deleting highlighted values in loyalty points form.	75
Figure 89 Before inserting values in loyalty points form.....	76
Figure 90 While inserting highlighted values in loyalty points form.	76
Figure 91 After inserting highlighted values in loyalty points form.	77
Figure 92 Before editing highlighted values in Delivery address form.	78
Figure 93 While editing highlighted values in Delivery address form.	78
Figure 94 After editing highlighted values in Delivery address form.	79
Figure 95 Before deleting highlighted values in Delivery address form.	80
Figure 96 After deleting highlighted values in Delivery address form.	80
Figure 97 Before inserting values in Delivery address form.....	81
Figure 98 While inserting highlighted values in Delivery address form.	82
Figure 99 After inserting highlighted values in Delivery address form.	83
Figure 100 Displaying customer order detail of aakash dahal.	84
Figure 101 Displaying order details of Samyak Sakhya.....	84
Figure 102 Displaying Veg Soup availability.	85
Figure 103 Displaying Chapatti availability.	85
Figure 104 Before editing value on Address id.	87

Figure 105 While editing value on Address id.....	87
Figure 106 After editing value on Address id.	88
Figure 107 Before editing value on telephone no.	89
Figure 108 While editing value on telephone no.	89
Figure 109 After editing value on telephone no.	90

Table 1 Actual table of example of Dishes.....	4
Table 2 Rearranged table of example of Dishes.....	4
Table 3 Rearranged example of Order Register.....	7
Table 4 Data dictionary of Loyalty_points table.	29
Table 5 Data dictionary of Dish table.	30
Table 6 Data dictionary of Restaurtant table.	31
Table 7 Data dictionary of Delivery_address table.....	32
Table 8 Data dictionary of Order_register table.	33
Table 9 Data dictionary of Customer table.....	34
Table 10 Data dictionary of Restaurant_dish table.....	35
Table 11 Data dictionary of Order table.	36
Table 12 Data dictionary of Dish_order table.....	37
Table 13 Order history is displayed.....	86

1. Introduction

In this coursework, we are assigned to create a web-based database application based on a given business case study. This coursework is to be built foremost to assess our evaluation on the design and development of the database system as well as to gain practical problem-solving skills. We are required to analyse, design and implement a web-based database application with an appropriate documentation. The web-based application is to be developed using Oracle SQL Developer, Data Modeler and ASP.NET with c#.

2. Textual Analysis and Preparation of Initial ERD

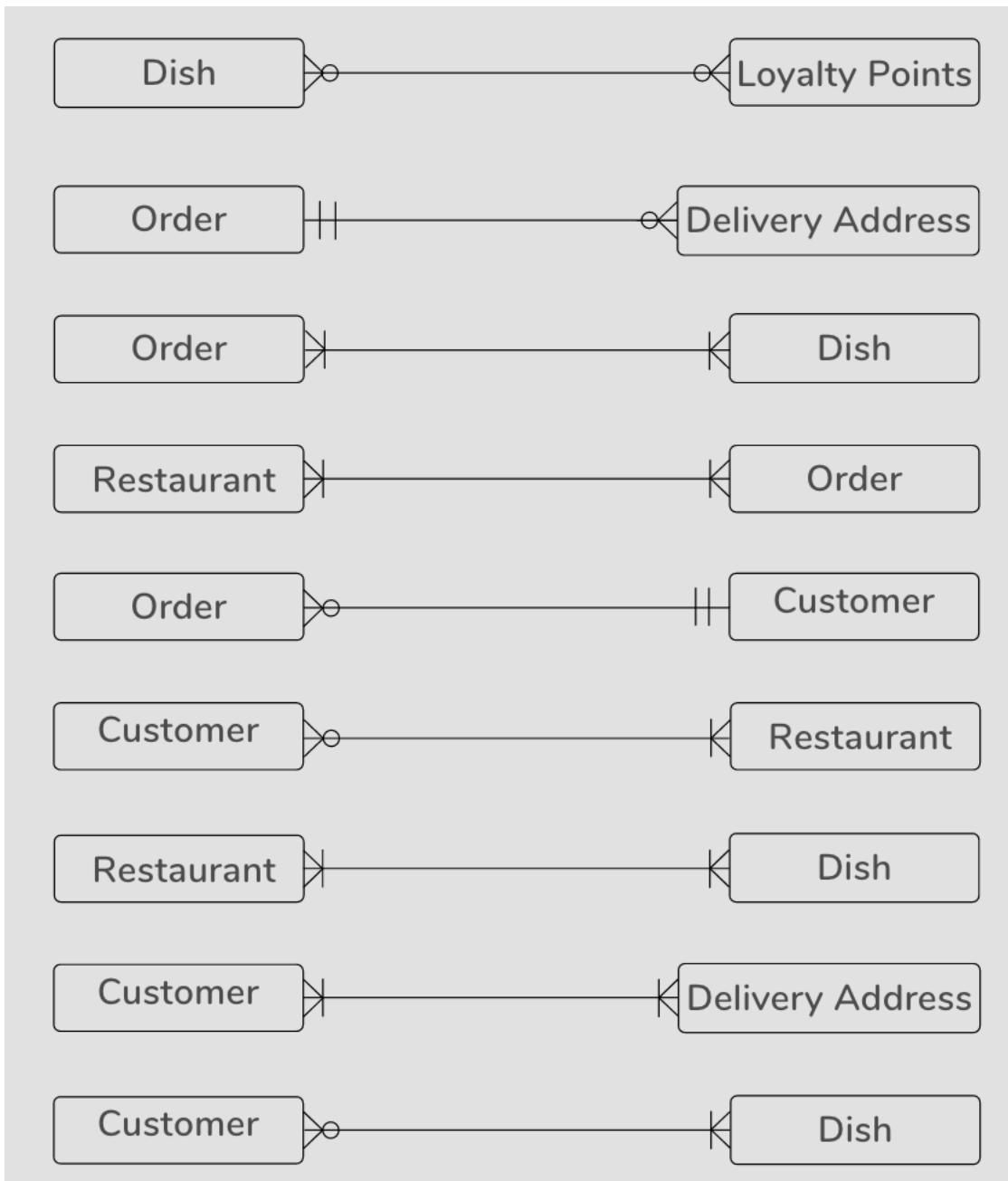


Figure 1 Individual entity relationship.

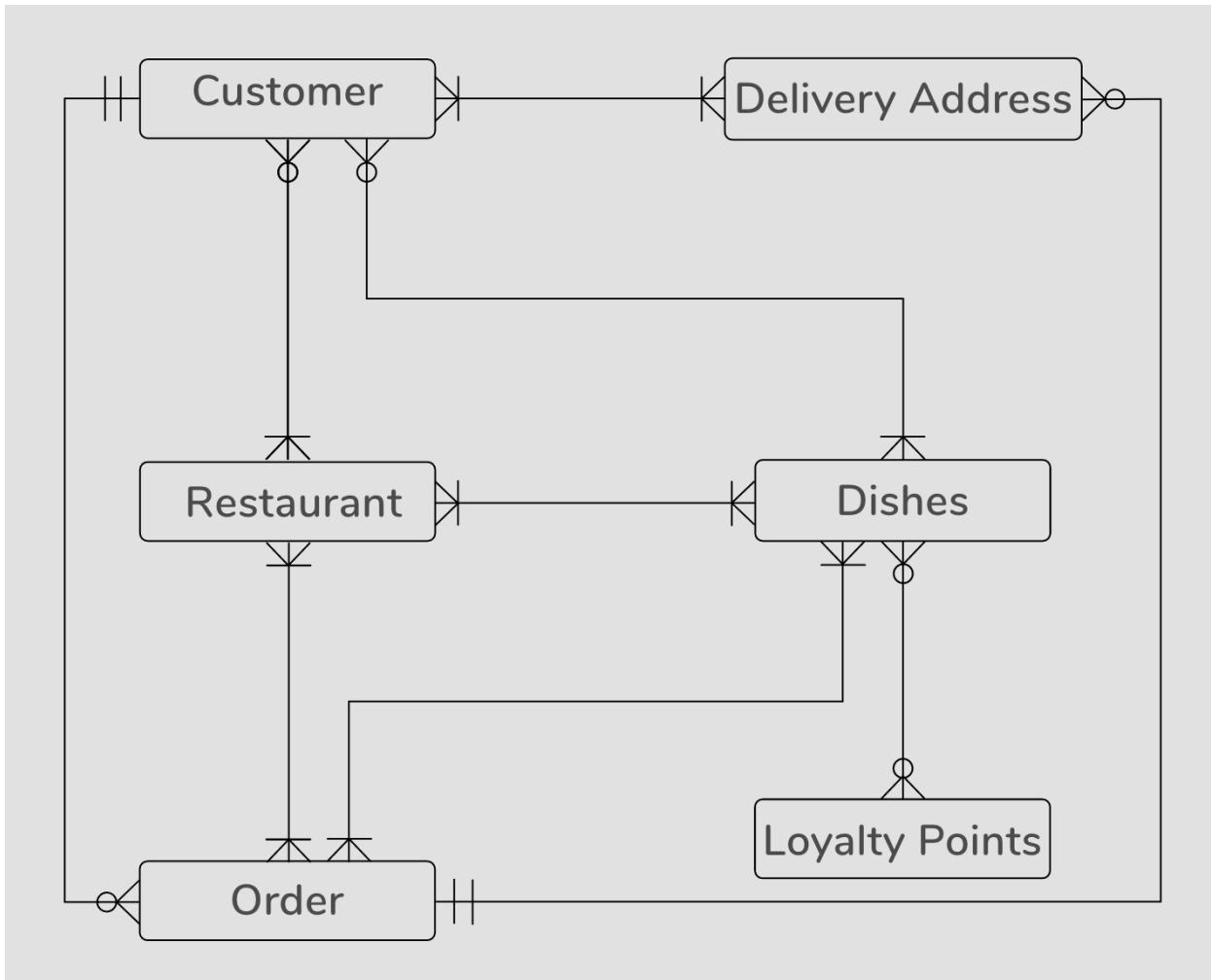


Figure 2 Complete initial ERD.

3. Normalization

3.1 Normalization of Fig 1

Actual table

Dish_code	Dish_name	Local_name	Available_on_restaurant
CHKMO	Chicken Momo	Chicken Momo	Royale Luxury
VEGTM	Vegetable Tempura		Royale Luxury
CHKMO	Chicken Momo	Chicken Momo	Taste Buds
MIXCHT	Mixed Chatamari	Nepali Pizza	Local Foodie

Table 1 Actual table of example of Dishes.

Rearranged table

Dish_code	Dish_name	Local_name	Available_on_restaurant
CHKMO	Chicken Momo	Chicken Momo	Royale Luxury
			Taste Buds
VEGTM	Vegetable Tempura		Royale Luxury
MIXCHT	Mixed Chatamari	Nepali Pizza	Local Foodie

Table 2 Rearranged table of example of Dishes.

UNF

In unnormalized form, repeating group are identified. Available_on_restaurant is the only repeating group as it has the multiple records of the restaurant for a single row of CHKMO.

Dish (Dish_code, Dish_name, Local_name, {Available_on_restaurant})

1NF

In 1st Normalized form, the repeating group are seperated.

Dish (Dish_code, Dish_name, Local_name)

Restaurant_available (Restaurant_id, Available_on_restaurant, Telephone_no
Dish_code*)

assumption: restaurant name must be unique.

2NF

In 2nd normalized form, the partial dependency is eliminated. we do not have any non-key attributes as partial dependency only occurs when a non-key attribute is functionally dependent on part of a candidate key. so, the entities are already in 2NF.

Dish (Dish_code, Dish_name, Local_name)

Restaurant_available (Restaurant_id, Available_on_restaurant, Telephone_no
Dish_code*)

3NF

In 3rd normalized form, the transitive dependency is eliminated. There is not any transitive dependency as there is no other non-key attribute.

Dish (Dish_code, Dish_name, Local_name)

Restaurant (Restaurant_id, Available_on_restaurant, Telephone_no)

Restaurant_dish (Restaurant_id*, Dish_code*)

3.2 Normalization of Fig 2

SN	Date	Time	Order no	Order amount	Delivery Address	Status	Dish code	Dish name	Order unit	Dish rate	Line total	Restaurant
2020-09223334	9th November 2020	14:30	09-00231	2750	Baneshwor	On time	CHKMO	Chicken momo	6 plates	350	2100	Taste buds
							MIXCHT	Mixed chatamari	2 plates	325	650	Local foodie
		14:40	09-00360	5000	baluwatar	On-time						
		16:50	09-00000	3450	Kalanki	Late						
2020-09223374	9th November 2020	15:20	09-00345	700	kalanki	Early	CHKMO	Chicken momo	2 plates	350	700	Taste buds

Table 3 Rearranged example of Order Register.

UNF

In unnormalized form, repeating group are identified.

Good Food (SN, Date, {time, order_no, order_amount, delivery_Address, status, {Dish code, Dish name, Order_unit, dish_rate, Line_total, restaurant}})

1NF

In 1st Normalized form, the repeating group are seperated.

Order_register (SN, Date)

Order (Order_no, Time, Order_amount, Delivery_Address, Status, SN*)

Dish (Dish_code, Dish_name, Order_unit, Dish_rate, Line_total, Restaurant, Order_no*)

2NF

In 2nd normalized form, the partial dependency is eliminated.

Order_no -> Time, Order_amount, Delivery_Address, Status

Dish_code-> Dish_name, Dish_rate, Restaurant

Dish_code, Order_no-> Order_unit, Line_total

Order_register (SN, Date)

Order (Order_no, Time, Order_amount, Delivery_Address, Status, SN*)

Dish (Dish_code, Dish_name, Dish_rate, Restaurant)

Dish_order (Dish_code*, Order_no*, Order_unit, Line_total)

3NF

In 3rd normalized form, the transitive dependency is eliminated.

Order_register (SN, Date)

Order (Order_no, Time, Order_amount, Delivery_Address, Status, SN*)

Dish (Dish_code, Dish_name, Dish_rate, Restaurant)

Dish_order (Dish_code*, Order_no*, Order_unit, Line_total)

4. Assumptions and Integration

- Address table is separated and created with all the attributes. Where the attributes are Address_Id, Longitude, latitude, province, district, city and ward.
- Delivery address is created as address for delivery was implemented in first initial ERD and it needs separate entity in final ERD form
- Restaurant_id is used as foreign key in many entity to get the details of particular restaurant.
- Loyalty table provides the points for different dish on different days.
- Addresss is used as foreign key in coustomer entity as it provides the delivery points.

5. Overall Entities

- Loyalty_points
 - Loyalty_id (PK)
 - Day
 - Loyalty_point
- Dish
 - Dish_code (PK)
 - Dish_name
 - Local_name
 - Dish_rate
 - Loyalty_id (FK)
- Restaurant
 - Restaurant_id (PK)
 - Available_on_restaurant
 - Telephone_no
- Restaurant_dish
 - Restaurant_id (PK)(FK)
 - Dish_code (PK)(FK)

- Delivery_address
 - Address_id (PK)
 - Latitude
 - Longitude
 - Province
 - City
 - Ward
- Customer
 - Customer_id (PK)
 - Customer_name
 - Contact
 - Address_id (FK)
- Order_register
 - SN (PK)
 - Date
- Order
 - Order_no (PK)
 - Time
 - Order_amount
 - Status

- SN (FK)
 - Customer_id (FK)
-
- Dish_order
 - Dish_code (PK)(FK)
 - Order_no (PK)(FK)
 - Order_unit
 - Line_total
 - Restaurant_id (FK)

6. ER-Model

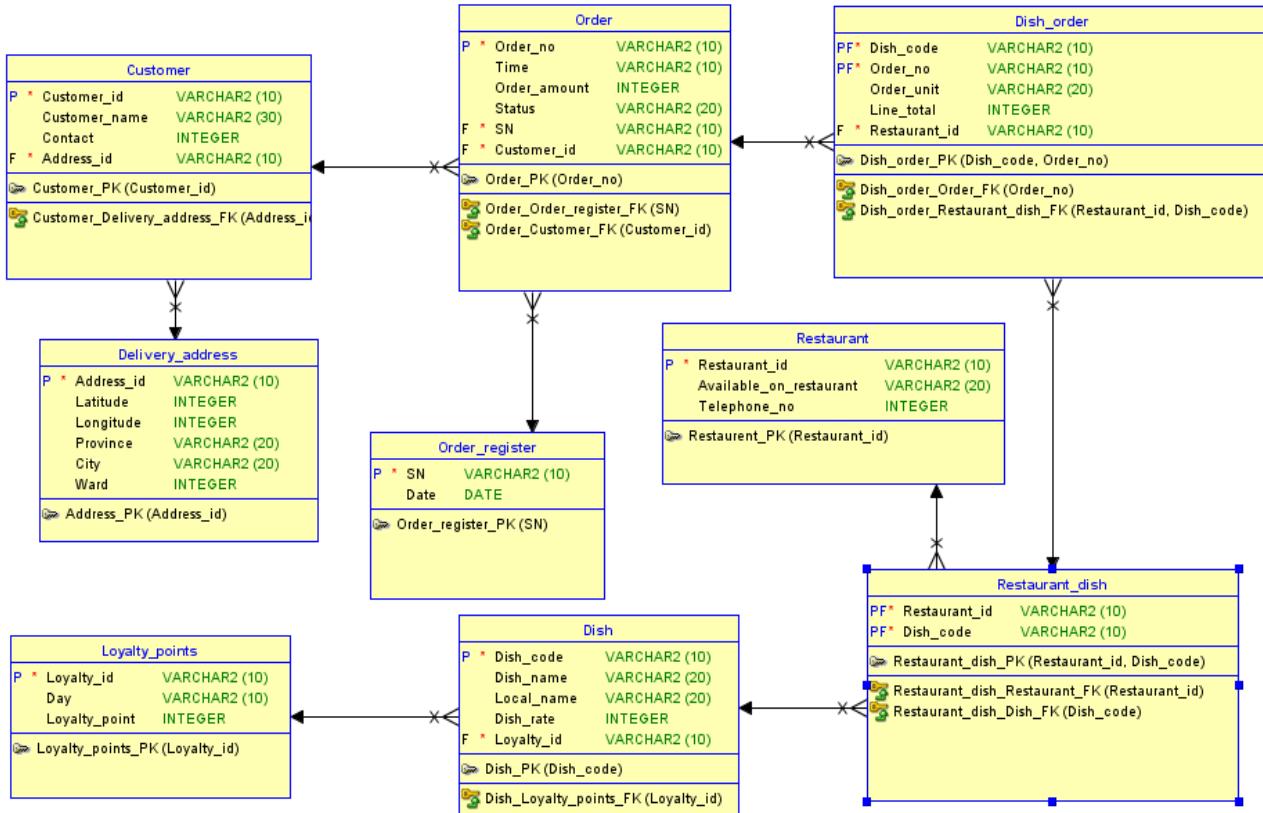


Figure 3 Final ERD.

7. Script

 DDL Preview

```

1 CREATE TABLE loyalty_points (
2     loyalty_id      VARCHAR2(10) NOT NULL,
3     day             VARCHAR2(10),
4     loyalty_point  INTEGER
5 );
6
7 ALTER TABLE loyalty_points ADD CONSTRAINT loyalty_points_pk PRIMARY KEY ( loyalty_id );

```

Figure 4 DDL script of **Loyalty_points** generated from DATA Modeler.

 DDL Preview

```

1 CREATE TABLE dish (
2     dish_code      VARCHAR2(10) NOT NULL,
3     dish_name      VARCHAR2(20),
4     local_name     VARCHAR2(20),
5     dish_rate      INTEGER,
6     loyalty_id    VARCHAR2(10) NOT NULL
7 );
8
9 ALTER TABLE dish ADD CONSTRAINT dish_pk PRIMARY KEY ( dish_code );
10
11 ALTER TABLE dish
12     ADD CONSTRAINT dish_loyalty_points_fk FOREIGN KEY ( loyalty_id )
13         REFERENCES loyalty_points ( loyalty_id )
14         ON DELETE CASCADE;

```

Figure 5 DDL script of **dish** generated from DATA Modeler.

 DDL Preview

```
1 | CREATE TABLE delivery_address (
2 |   address_id  VARCHAR2(10) NOT NULL,
3 |   latitude     INTEGER,
4 |   longitude    INTEGER,
5 |   province     VARCHAR2(20),
6 |   city         VARCHAR2(20),
7 |   ward         INTEGER
8 | );
9 |
10| ALTER TABLE delivery_address ADD CONSTRAINT address_pk PRIMARY KEY ( address_id );
```

Figure 6 DDL script of **delivery_address** generated from DATA Modeler.

 DDL Preview

```
1 | CREATE TABLE order_register (
2 |   sn      VARCHAR2(10) NOT NULL,
3 |   "Date"  DATE
4 | );
5 |
6 | ALTER TABLE order_register ADD CONSTRAINT order_register_pk PRIMARY KEY ( sn );
```

Figure 7 DDL script of **order_register** generated from DATA Modeler.

 DDL Preview

```
1 |CREATE TABLE restaurant (
2 |    restaurant_id      VARCHAR2(10) NOT NULL,
3 |    available_on_restaurant VARCHAR2(20),
4 |    telephone_no        INTEGER
5 |);
6 |
7 |ALTER TABLE restaurant ADD CONSTRAINT restaurent_pk PRIMARY KEY ( restaurant_id );
```

Figure 8 DDL script of **restaurant** generated from DATA Modeler.

 DDL Preview

```
1 |CREATE TABLE customer (
2 |    customer_id      VARCHAR2(10) NOT NULL,
3 |    customer_name    VARCHAR2(30),
4 |    contact          INTEGER,
5 |    address_id       VARCHAR2(10) NOT NULL
6 |);
7 |
8 |ALTER TABLE customer ADD CONSTRAINT customer_pk PRIMARY KEY ( customer_id );
9 |
10|ALTER TABLE customer
11|    ADD CONSTRAINT customer_delivery_address_fk FOREIGN KEY ( address_id )
12|        REFERENCES delivery_address ( address_id )
13|            ON DELETE CASCADE;
```

Figure 9 DDL script of **customer** generated from DATA Modeler.

 DDL Preview

```

1  CREATE TABLE restaurant_dish (
2      restaurant_id  VARCHAR2(10) NOT NULL,
3      dish_code      VARCHAR2(10) NOT NULL
4  );
5
6  ALTER TABLE restaurant_dish ADD CONSTRAINT restaurant_dish_pk PRIMARY KEY ( restaurant_id,
7                                         dish_code );
8
9  ALTER TABLE restaurant_dish
10    ADD CONSTRAINT restaurant_dish_dish_fk FOREIGN KEY ( dish_code )
11        REFERENCES dish ( dish_code )
12        ON DELETE CASCADE;
13
14 ALTER TABLE restaurant_dish
15    ADD CONSTRAINT restaurant_dish_restaurant_fk FOREIGN KEY ( restaurant_id )
16        REFERENCES restaurant ( restaurant_id )
17        ON DELETE CASCADE;

```

Figure 10 DDL script of **restaurent_dish** generated from DATA Modeler.

 DDL Preview

```

1  CREATE TABLE "Order" (
2      order_no      VARCHAR2(10) NOT NULL,
3      time         VARCHAR2(10),
4      order_amount INTEGER,
5      status        VARCHAR2(20),
6      sn            VARCHAR2(10) NOT NULL,
7      customer_id  VARCHAR2(10) NOT NULL
8  );
9
10 ALTER TABLE "Order" ADD CONSTRAINT order_pk PRIMARY KEY ( order_no );
11
12 ALTER TABLE "Order"
13    ADD CONSTRAINT order_customer_fk FOREIGN KEY ( customer_id )
14        REFERENCES customer ( customer_id )
15        ON DELETE CASCADE;
16
17 ALTER TABLE "Order"
18    ADD CONSTRAINT order_order_register_fk FOREIGN KEY ( sn )
19        REFERENCES order_register ( sn )
20        ON DELETE CASCADE;

```

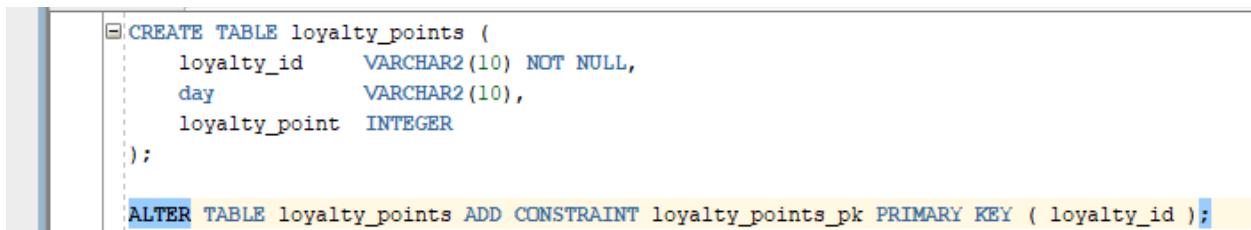
Figure 11 DDL script of **order** generated from DATA Modeler.

DDL Preview

```
1 CREATE TABLE dish_order (
2     dish_code      VARCHAR2(10) NOT NULL,
3     order_no       VARCHAR2(10) NOT NULL,
4     order_unit     VARCHAR2(20),
5     line_total     INTEGER,
6     restaurant_id  VARCHAR2(10) NOT NULL
7 );
8
9 ALTER TABLE dish_order ADD CONSTRAINT dish_order_pk PRIMARY KEY ( dish_code,
10                                         order_no );
11
12 ALTER TABLE dish_order
13     ADD CONSTRAINT dish_order_order_fk FOREIGN KEY ( order_no )
14         REFERENCES "Order" ( order_no )
15         ON DELETE CASCADE;
16
17 ALTER TABLE dish_order
18     ADD CONSTRAINT dish_order_restaurant_dish_fk FOREIGN KEY ( restaurant_id,
19                                         dish_code )
20         REFERENCES restaurant_dish ( restaurant_id,
21                                         dish_code )
22         ON DELETE CASCADE;
```

Figure 12 DDL script of *dish_order* generated from DATA Modeler.

8. Script Execution



```
CREATE TABLE loyalty_points (
    loyalty_id      VARCHAR2(10) NOT NULL,
    day             VARCHAR2(10),
    loyalty_point  INTEGER
);

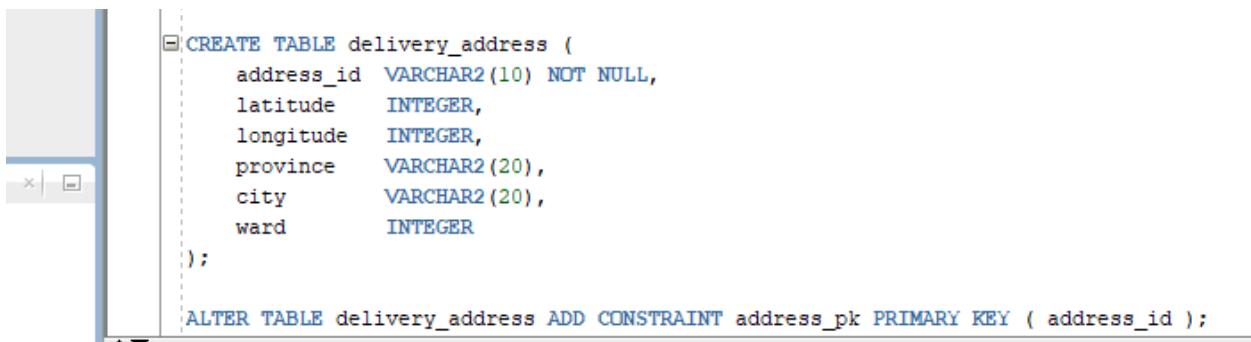
ALTER TABLE loyalty_points ADD CONSTRAINT loyalty_points_pk PRIMARY KEY ( loyalty_id );
```

Figure 13 Execution of DDL generated script of **loyalty_points**.

Table LOYALTY_POINTS created.

Table LOYALTY_POINTS altered.

Figure 14 DDL generated script of **loyalty_points** is executed, and table is created successfully.

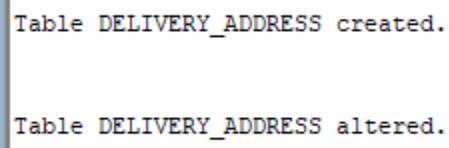


The screenshot shows a SQL developer interface with a code editor window. The code contains two statements: a CREATE TABLE statement defining columns for address_id, latitude, longitude, province, city, and ward, and an ALTER TABLE statement adding a primary key constraint named address_pk on the address_id column.

```
CREATE TABLE delivery_address (
    address_id  VARCHAR2(10) NOT NULL,
    latitude     INTEGER,
    longitude    INTEGER,
    province    VARCHAR2(20),
    city        VARCHAR2(20),
    ward         INTEGER
);

ALTER TABLE delivery_address ADD CONSTRAINT address_pk PRIMARY KEY ( address_id );
```

Figure 15 Execution of DDL generated script of **delivery_address**.



The screenshot shows the results of the executed DDL script. It displays two messages: "Table DELIVERY_ADDRESS created." indicating the successful creation of the table, and "Table DELIVERY_ADDRESS altered." indicating the successful addition of the primary key constraint.

Table DELIVERY_ADDRESS created.
Table DELIVERY_ADDRESS altered.

Figure 16 DDL generated script of **delivery_address** is executed, and table is created successfully.

The screenshot shows a SQL editor window in Oracle SQL Developer. The code consists of two statements:

```
CREATE TABLE order_register (
    sn      VARCHAR2(10) NOT NULL,
    "Date"  DATE
);

ALTER TABLE order_register ADD CONSTRAINT order_register_pk PRIMARY KEY ( sn );
```

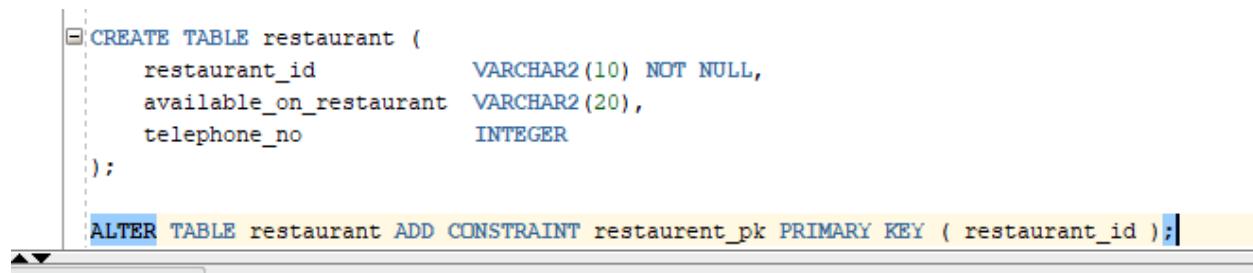
The second statement, `ALTER TABLE order_register ADD CONSTRAINT order_register_pk PRIMARY KEY (sn);`, is highlighted in blue, indicating it is the current statement being executed.

Figure 17 Execution of DDL generated script of **order_register**.

Table ORDER_REGISTER created.

Table ORDER_REGISTER altered.

Figure 18 DDL generated script of **order_register** is executed, and table is created successfully.

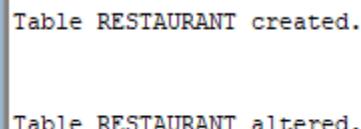


The screenshot shows a SQL editor window in Oracle SQL Developer. The code entered is:

```
CREATE TABLE restaurant (
    restaurant_id      VARCHAR2(10) NOT NULL,
    available_on_restaurant VARCHAR2(20),
    telephone_no       INTEGER
);

ALTER TABLE restaurant ADD CONSTRAINT restaurent_pk PRIMARY KEY ( restaurant_id );
```

Figure 19 Execution of DDL generated script of **restaurant**.



The screenshot shows the output of the executed DDL statements:

```
Table RESTAURANT created.  
Table RESTAURANT altered.
```

Figure 20 DDL generated script of **restaurant** is executed, and table is created successfully.

```
CREATE TABLE dish (
    dish_code  VARCHAR2(10) NOT NULL,
    dish_name  VARCHAR2(20),
    local_name VARCHAR2(20),
    dish_rate   INTEGER,
    loyalty_id VARCHAR2(10) NOT NULL
);

ALTER TABLE dish ADD CONSTRAINT dish_pk PRIMARY KEY ( dish_code );

ALTER TABLE dish
    ADD CONSTRAINT dish_loyalty_points_fk FOREIGN KEY ( loyalty_id )
        REFERENCES loyalty_points ( loyalty_id )
        ON DELETE CASCADE;
```

Figure 21 Execution of DDL generated script of **dish**.

```
Table DISH created.

Table DISH altered.

Table DISH altered.
```

Figure 22 DDL generated script of **dish** is executed, and table is created successfully.

```
CREATE TABLE customer (
    customer_id      VARCHAR2(10) NOT NULL,
    customer_name    VARCHAR2(30),
    contact          INTEGER,
    address_id       VARCHAR2(10) NOT NULL
);

ALTER TABLE customer ADD CONSTRAINT customer_pk PRIMARY KEY ( customer_id );

ALTER TABLE customer
    ADD CONSTRAINT customer_delivery_address_fk FOREIGN KEY ( address_id )
        REFERENCES delivery_address ( address_id )
        ON DELETE CASCADE;
```

Figure 23 Execution of DDL generated script of **customer**.

```
Table CUSTOMER created.

Table CUSTOMER altered.

Table CUSTOMER altered.
```

Figure 24 DDL generated script of **customer** is executed, and table is created successfully.

```
CREATE TABLE restaurant_dish (
    restaurant_id  VARCHAR2(10) NOT NULL,
    dish_code      VARCHAR2(10) NOT NULL
);

ALTER TABLE restaurant_dish ADD CONSTRAINT restaurant_dish_pk PRIMARY KEY ( restaurant_id,
                                                               dish_code );

ALTER TABLE restaurant_dish
    ADD CONSTRAINT restaurant_dish_dish_fk FOREIGN KEY ( dish_code )
        REFERENCES dish ( dish_code )
        ON DELETE CASCADE;

ALTER TABLE restaurant_dish
    ADD CONSTRAINT restaurant_dish_restaurant_fk FOREIGN KEY ( restaurant_id )
        REFERENCES restaurant ( restaurant_id )
        ON DELETE CASCADE;
```

Figure 25 Execution of DDL generated script of **restaurant_dish**.

```
Table RESTAURANT_DISH created.

Table RESTAURANT_DISH altered.

Table RESTAURANT_DISH altered.

Table RESTAURANT_DISH altered.
```

Figure 26 DDL generated script of **restaurant_dish** is executed, and table is created successfully.

```
CREATE TABLE "Order" (
    order_no      VARCHAR2(10) NOT NULL,
    time          VARCHAR2(10),
    order_amount  INTEGER,
    status         VARCHAR2(20),
    sn            VARCHAR2(10) NOT NULL,
    customer_id   VARCHAR2(10) NOT NULL
);

ALTER TABLE "Order" ADD CONSTRAINT order_pk PRIMARY KEY ( order_no );

ALTER TABLE "Order"
    ADD CONSTRAINT order_customer_fk FOREIGN KEY ( customer_id )
        REFERENCES customer ( customer_id )
        ON DELETE CASCADE;

ALTER TABLE "Order"
    ADD CONSTRAINT order_order_register_fk FOREIGN KEY ( sn )
        REFERENCES order_register ( sn )
        ON DELETE CASCADE;
```

Figure 27 Execution of DDL generated script of **order**.

```
Table "Order" created.

Table "Order" altered.

Table "Order" altered.

Table "Order" altered.
```

Figure 28 DDL generated script of **order** is executed, and table is created successfully.

```
CREATE TABLE dish_order (
    dish_code      VARCHAR2(10) NOT NULL,
    order_no       VARCHAR2(10) NOT NULL,
    order_unit     VARCHAR2(20),
    line_total     INTEGER,
    restaurant_id  VARCHAR2(10) NOT NULL
);

ALTER TABLE dish_order ADD CONSTRAINT dish_order_pk PRIMARY KEY ( dish_code,
    order_no );

ALTER TABLE dish_order
    ADD CONSTRAINT dish_order_order_fk FOREIGN KEY ( order_no )
        REFERENCES "Order" ( order_no )
        ON DELETE CASCADE;

ALTER TABLE dish_order
    ADD CONSTRAINT dish_order_restaurant_dish_fk FOREIGN KEY ( restaurant_id,
        dish_code )
        REFERENCES restaurant_dish ( restaurant_id,
            dish_code )
        ON DELETE CASCADE;
```

Figure 29 Execution of DDL generated script of **dish_order**.

```
Table DISH_ORDER created.

Table DISH_ORDER altered.

Table DISH_ORDER altered.

Table DISH_ORDER altered.
```

Figure 30 DDL generated script of **dish_order** is executed, and table is created successfully.

9. Data Dictionary

Table Name: **Loyalty_points**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Loyalty_id	VARCHAR2	10	Loyalty_points_PK			Unique code for loyalty point.	L1
Day	VARCHAR2	10				Days in a week.	Friday
Loyalty_point	INTEGER					Points for loyalty.	10

Table 4 Data dictionary of Loyalty_points table.

Table Name: **Dish**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Dish_code	VARCHAR2	10	Dish_PK			Unique code for dish.	CHPT
Dish_name	VARCHAR2	20				Name of the dish.	Chapatti
Local_name	VARCHAR2	20				Local name of the dish.	roti
Dish_rate	INTEGER					Cost of the dish.	50
Loyalty_id	VARCHAR2	10	Dish_Loyalty_point_s_FK	Loyalty_points	Loyalty_id	Referencing loyalty_id for the loyalty points.	L2

Table 5 Data dictionary of Dish table.

Table Name: **Restaurant**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Restaurant_id	VARCHAR2	10	Restaurant_PK			Unique code for Restaurant.	R1
Available_on_restaurant	VARCHAR2	20				Restaurant name.	Momo Vibe
Telephone_no	INTEGER					Telephone number of the restaurant.	555555

Table 6 Data dictionary of Restaurant table.

Table Name: **Delivery_address**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Address_id	VARCHAR2	10	Address_PK			Unique code for Address.	A1
Latitude	INTEGER					Latitude of the address.	29.4
Longitude	INTEGER					Longitude of the address.	85.13
Province	VARCHAR2	20				Province where the address falls on.	Province 1
City	VARCHAR2	20				City the address falls on.	Urlabari
Ward	INTEGER					Ward number of the city.	1

Table 7 Data dictionary of Delivery_address table.

Table Name: **Order_register**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
SN	VARCHAR2	10	Order_register_PK			Unique code for order registration.	21-O1
Date	DATE					Date of the order registration.	13/12/2021

Table 8 Data dictionary of Order_register table.

Table Name: **Customer**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Customer_id	VARCHAR2	10	Customer _PK			Unique code for customer.	C1
Customer_name	VARCHAR2	30				Name of the customer.	Girwan Budhath oki
Contact	INTEGER					Contact number of the customer.	9842154 326
Address_id	VARCHAR2	10	Customer _Delivery_address _FK	Delivery_address	Address_id	Address of the Customer.	A5

Table 9 Data dictionary of Customer table.

Table Name: **Restaurant_dish**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Restaurant_id	VARCHAR2	10	Restaurant_dish_PK, Restaurant_dish_restaurant_FK	Restaurant	Restaurant_id	Code of the restaurant where the dish is available.	R6
Dish_code	VARCHAR2	10	Restaurant_dish_PK, Restaurant_dish_Dish_FK	Dish	Dish_code	Code for the dish.	CKPZA

Table 10 Data dictionary of Restaurant_dish table.

Table Name: **Order**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Order_no	VARCHAR2	10	Order_PK			Unique code for Order.	O-1
Time	VARCHAR2	10				Time of the order.	1:01
Order_amount	INTEGER					Amount of the order.	1100
Status	VARCHAR2	20				Status of the order delivery	late
SN	VARCHAR2	10	Order_Or der_regis ter_FK	Order_regs ter	SN	Unique code of the order registration.	21-O5
Customer_id	VARCHAR2	10	Order_Cu stomer_F K	Customer	Customer_id	Unique code of the customer.	C7

Table 11 Data dictionary of Order table.

Table Name: **Dish_order**

Attributes	Data Type	Size	Constraints	Reference Table (if Foreign Key)	Reference Column (if Foreign Key)	Description	Example Data
Dish_code	VARCHAR2	10	Dish_order_PK, Dish_order_Restaurant_dish_FK	Restaurant	Dish_code	Unique code for dish.	CKNUG
Order_no	VARCHAR2	10	Dish_order_PK, Dish_order_Order_FK	Order	Order_no	Unique code for order.	O-9
Order_unit	VARCHAR2	20				Quantity of dish ordered.	1 Plate
Line_total	INTEGER					Amount for dish times quantity.	250
Restaurant_id	VARCHAR2	10	Dish_order_Restaurant_dish_FK	Restaurant	Restaurant_id	Unique code of the restaurant.	R1

Table 12 Data dictionary of Dish_order table

10. INSERT Statement

```
insert into Loyalty_points values('L1','Wednesday','10');
insert into Loyalty_points values('L2','Thursday','5');
insert into Loyalty_points values('L3','Friday','15');
insert into Loyalty_points values('L4','Saturday','20');
insert into Loyalty_points values('L5','Sunday','10');
insert into Loyalty_points values('L6','Monday','10');
insert into Loyalty_points values('L7','Tuesday','5');
insert into Loyalty_points values('L8','Wednesday','15');
insert into Loyalty_points values('L9','Thursday','20');
insert into Loyalty_points values('L10','Friday','5');
```

Figure 31 Insertion of statements in **Loyalty_points** table.

```
1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.
```

Figure 32 Data are inserted in **loyalty_points** table successfully.

```
insert into Dish values('VGSUP','Veg Soup','null','50','L1');
insert into Dish values('CKMO','Chicken Momo','Chicken Momo','100','L1');
insert into Dish values('CHPT','Chapatti','roti','50','L2');
insert into Dish values('T','Tea','chiya','30','L4');
insert into Dish values('CKNUG','Chicken Nuggets','null','250','L4');
insert into Dish values('CKSEK','Chicken Sekuwa','null','200','L7');
insert into Dish values('VGBGR','Veg Burger','null','80','L8');
insert into Dish values('BFSKT','Buff Sukuti','buff sukuti','250','L2');
insert into Dish values('CKPZA','Chicken Pizza','null','300','L10');
insert into Dish values('VGPZA','Veg Pizza','null','220','L3');
```

Figure 33 Insertion of statements in **Dish** table.

```
1 row inserted.  
1 row inserted.  
1 row inserted.  
1 row inserted.
```

Figure 34 Data are inserted in **Dish** table successfully.

```
insert into Restaurant values('R1','Restaurant Nepal','555111');
insert into Restaurant values('R2','Kitchen Urlabari','555222');
insert into Restaurant values('R3','Nepali Restaurant','555333');
insert into Restaurant values('R4','Test Of Nepal','555444');
insert into Restaurant values('R5','Momo Vibe','555555');
insert into Restaurant values('R6','Nepalese Flavor','555666');
insert into Restaurant values('R7','Grill Center','555777');
insert into Restaurant values('R8','Sekuwa Corner','555888');
insert into Restaurant values('R9','Sigma','555999');
insert into Restaurant values('R10','Boudha Restro','555110');
```

Figure 35 Insertion of statements in **Restaurant** table.

```
1 row inserted.
```

```
1 row inserted.
```

```
1 row inserted.
```

Figure 36 Data are inserted in **Loyal** table successfully.

```
insert into Delivery_address values('A1','28.39','84.12','Province 1','Urlabari','1');
insert into Delivery_address values('A2','29.4','85.13','Province 1','Urlabari','2');
insert into Delivery_address values('A3','30.41','86.14','Province 1','Urlabari','3');
insert into Delivery_address values('A4','31.42','87.15','Province 1','Urlabari','4');
insert into Delivery_address values('A5','32.43','88.16','Province 1','Urlabari','5');
insert into Delivery_address values('A6','33.44','89.17','Province 1','Damak','6');
insert into Delivery_address values('A7','34.45','90','Province 1','Damak','7');
insert into Delivery_address values('A8','35.46','90.83','Province 1','Damak','8');
insert into Delivery_address values('A9','36.47','91.66','Province 1','Damak','9');
insert into Delivery_address values('A10','37.48','92.49','Province 1','Damak','10');
```

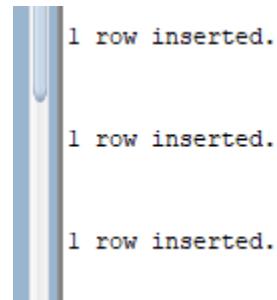
Figure 37 Insertion of statements in **Delivery_address** table.

```
1 row inserted.  
1 row inserted.  
1 row inserted.
```

Figure 38 Data are inserted in **Delivery_address** table successfully.

```
insert into Order_register values('21-01', TO_DATE('13/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-02', TO_DATE('14/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-03', TO_DATE('13/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-04', TO_DATE('20/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-05', TO_DATE('16/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-06', TO_DATE('22/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-07', TO_DATE('13/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-08', TO_DATE('14/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-09', TO_DATE('16/12/2021','DD/MM/YYYY'));
insert into Order_register values('21-010', TO_DATE('14/12/2021','DD/MM/YYYY'));
```

Figure 39 Insertion of statements in *Order_register* table.

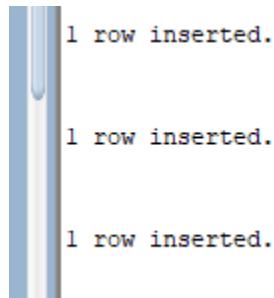


```
1 row inserted.  
1 row inserted.  
1 row inserted.
```

Figure 40 Data are inserted in *Order_register* table successfully.

```
insert into Customer values('C1','Girwan Budhathoki','9842154326','A5');
insert into Customer values('C2','Ram Rawal','9842154437','A7');
insert into Customer values('C3','Sabin Mukhia','9842154548','A9');
insert into Customer values('C4','Dilip Pun','9842154659','A3');
insert into Customer values('C5','Kabi Darlami','9842154770','A4');
insert into Customer values('C6','Ujwal Risal ','9842154881','A9');
insert into Customer values('C7','Samyak Sakhya','9842154992','A1');
insert into Customer values('C8','Sumit Subba','9842155103','A2');
insert into Customer values('C9','Raju Shrestha','9842155214','A8');
insert into Customer values('C10','Dipesh Bhandari','9842155325','A3');
```

Figure 41 Insertion of statements in **Customer** table.



```
1 row inserted.  
1 row inserted.  
1 row inserted.
```

Figure 42 Data are inserted in **Customer** table successfully.

```
insert into Restaurant_dish values('R9','VGSUP');
insert into Restaurant_dish values('R5','CKMO');
insert into Restaurant_dish values('R5','CHPT');
insert into Restaurant_dish values('R7','T');
insert into Restaurant_dish values('R1','CKNUG');
insert into Restaurant_dish values('R1','VGSUP');
insert into Restaurant_dish values('R9','CKMO');
insert into Restaurant_dish values('R3','CHPT');
insert into Restaurant_dish values('R6','CKPZA');
insert into Restaurant_dish values('R7','VGPZA');
insert into Restaurant_dish values('R8','VGSUP');
insert into Restaurant_dish values('R10','CKMO');
insert into Restaurant_dish values('R1','CHPT');
insert into Restaurant_dish values('R2','T');
insert into Restaurant_dish values('R6','CKNUG');
```

Figure 43 Insertion of statements in **Restaurant_dish** table.

```
1 row inserted.
```

```
1 row inserted.
```

```
1 row inserted.
```

Figure 44 Data are inserted in loyal **Restaurant_dish** successfully.

```
insert into "Order" values('0-1','1:01',1100,'late','21-01','C4');
insert into "Order" values('0-2','2:22',1560,'on time','21-08','C5');
insert into "Order" values('0-3','3:43',2020,'on time','21-09','C6');
insert into "Order" values('0-4','5:04',2480,'on time','21-05','C7');
insert into "Order" values('0-5','6:25',2940,'early','21-06','C8');
insert into "Order" values('0-6','7:46',3400,'early','21-02','C9');
insert into "Order" values('0-7','9:07',3860,'late','21-07','C2');
insert into "Order" values('0-8','10:28',4320,'on time','21-08','C3');
insert into "Order" values('0-9','11:49',4780,'on time','21-09','C4');
insert into "Order" values('0-10','17:13',5240,'late','21-05','C7');
insert into "Order" values('0-11','18:34',5700,'on time','21-06','C8');
insert into "Order" values('0-12','5:04',6160,'on time','21-02','C9');
insert into "Order" values('0-13','5:55',6620,'on time','21-03','C2');
insert into "Order" values('0-14','4:55',7080,'early','21-04','C3');
insert into "Order" values('0-15','5:32',7540,'early','21-05','C4');
```

Figure 45 Insertion of statements in **Order** table.

```
1 row inserted.
```

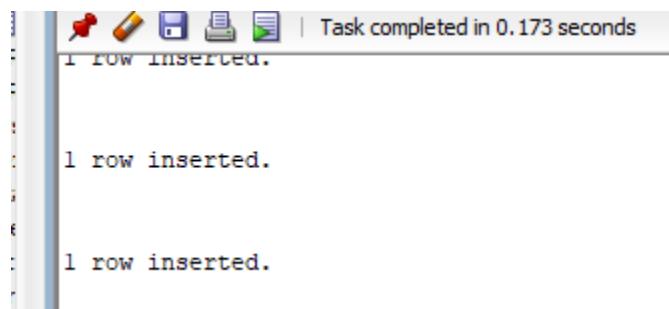
```
1 row inserted.
```

```
1 row inserted.
```

Figure 46 Data are inserted in **Order** table successfully.

```
insert into Dish_order values('VGSUP','0-3','1 plate','100','R9');
insert into Dish_order values('CKMO','0-6','2 plate','200','R5');
insert into Dish_order values('CHPT','0-7','3 plate','150','R5');
insert into Dish_order values('T','0-8','2 plate','60','R7');
insert into Dish_order values('CKNUG','0-9','1 plate','250','R1');
insert into Dish_order values('VGSUP','0-10','3 plate','300','R1');
insert into Dish_order values('CKMO','0-11','1 plate','100','R9');
insert into Dish_order values('CHPT','0-12','2 plate','100','R3');
insert into Dish_order values('CKPZA','0-13','1 plate','300','R6');
insert into Dish_order values('VGPZA','0-2','1 plate','220','R7');
insert into Dish_order values('VGSUP','0-3','1 plate','100','R8');
insert into Dish_order values('CKMO','0-4','2 plate','200','R10');
insert into Dish_order values('CHPT','0-5','2 plate','100','R1');
insert into Dish_order values('T','0-6','2 plate','60','R2');
insert into Dish_order values('CKNUG','0-7','1 plate','250','R6');
```

Figure 47 Insertion of statements in *Dish_order* table.



The screenshot shows a database interface with a toolbar at the top featuring icons for file, edit, and database operations. The main area displays a SQL command and its execution results. The command is identical to the one shown in Figure 47. The results show three separate insertions, each followed by a confirmation message: "1 row inserted." The total execution time is listed as 0.173 seconds.

```
Task completed in 0.173 seconds
1 row inserted.

1 row inserted.

1 row inserted.
```

Figure 48 Data are inserted in *Dish_order* table successfully.

11. Commit

The screenshot shows a database management interface. At the top, there is a code editor window containing the text "COMMIT;". Below it is a toolbar with various icons. A message window titled "Script Output" displays the text "Commit complete.".

Figure 49 Commit is completed successfully.

12. SELECT Statement

The screenshot shows the Oracle SQL Developer interface. In the top-left query editor, the SQL command `select * from loyalty_points;` is entered. Below the editor, the status bar indicates "All Rows Fetched: 10 in 0.004 seconds". The main window displays the results in a tabular format. The table has three columns: `LOYALTY_ID`, `DAY`, and `LOYALTY_POINT`. The data is as follows:

LOYALTY_ID	DAY	LOYALTY_POINT
1 L1	Wednesday	10
2 L2	Thursday	5
3 L3	Friday	15
4 L4	Saturday	20
5 L5	Sunday	10
6 L6	Monday	10
7 L7	Tuesday	5
8 L8	Wednesday	15
9 L9	Thursday	20
10 L10	Friday	5

Figure 50 All data of `loyalty_points` table is displayed.

A screenshot of a SQL query interface. The query in the top pane is:

```
select * from dish;
```

The results are displayed in a table below, with tabs for Script Output, Query Result, and SQL. The table has columns: DISH_CODE, DISH_NAME, LOCAL_NAME, DISH_RATE, and LOYALTY_ID. The data is as follows:

	DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID
1	VGSUP	Veg Soup	null	50	L1
2	CKMO	Chicken Momo	Chicken Momo	100	L1
3	CHPT	Chapatti	roti	50	L2
4	T	Tea	chiya	30	L4
5	CKNUG	Chicken Nuggets	null	250	L4
6	CKSEK	Chicken Sekuwa	null	200	L7
7	VGBGR	Veg Burger	null	80	L8
8	BFSKT	Buff Sukuti	buff sukuti	250	L2
9	CKPZA	Chicken Pizza	null	300	L10
10	VGPZA	Veg Pizza	null	220	L3

Figure 51 All data of **dish** table is displayed.

A screenshot of a SQL query interface. The query in the top pane is:

```
select * from restaurant;
```

The results are displayed in a table below, with tabs for Script Output, Query Result, and SQL. The table has columns: RESTAURANT_ID, AVAILABLE_ON_RESTAURANT, and TELEPHONE_NO. The data is as follows:

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
1	R1	Restaurant Nepal	555111
2	R2	Kitchen Urlabari	555222
3	R3	Nepali Restaurant	555333
4	R4	Test Of Nepal	555444
5	R5	Momo Vibe	555555
6	R6	Nepalese Flavor	555666
7	R7	Grill Center	555777
8	R8	Sekuwa Corner	555888
9	R9	Sigma	555999
10	R10	Boudha Restro	555110

Figure 52 All data of **restaurant** table is displayed.

The screenshot shows the SQL Server Management Studio interface. A query window is open with the following SQL command:

```
select * from delivery_address;
```

The results pane displays the data from the **delivery_address** table. The table has columns: ADDRESS_ID, LATITUDE, LONGITUDE, PROVINCE, CITY, and WARD. The data shows 10 rows of address entries, all belonging to Province 1.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
1	A1	28	84	Province 1	Urlabari	1
2	A2	29	85	Province 1	Urlabari	2
3	A3	30	86	Province 1	Urlabari	3
4	A4	31	87	Province 1	Urlabari	4
5	A5	32	88	Province 1	Urlabari	5
6	A6	33	89	Province 1	Damak	6
7	A7	34	90	Province 1	Damak	7
8	A8	35	91	Province 1	Damak	8
9	A9	36	92	Province 1	Damak	9
10	A10	37	92	Province 1	Damak	10

Figure 53 All data of **delivery_address** table is displayed.

The screenshot shows the SQL Server Management Studio interface. A query window is open with the following SQL command:

```
select * from order_register;
```

The results pane displays the data from the **order_register** table. The table has columns: SN and Date. The data shows 10 rows of order registration dates.

	SN	Date
1	21-01	13-DEC-21
2	21-02	14-DEC-21
3	21-03	13-DEC-21
4	21-04	20-DEC-21
5	21-05	16-DEC-21
6	21-06	22-DEC-21
7	21-07	13-DEC-21
8	21-08	14-DEC-21
9	21-09	16-DEC-21
10	21-010	14-DEC-21

Figure 54 All data of **order_register** table is displayed.

The screenshot shows a database interface with a query editor and a results viewer. The query editor contains the SQL command: `select * from customer;`. The results viewer, titled "Query Result", displays the data from the customer table. The table has four columns: CUSTOMER_ID, CUSTOMER_NAME, CONTACT, and ADDRESS_ID. The data consists of 10 rows, each representing a customer with a unique ID, name, contact number, and address ID.

	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
1	C1	Girwan Budhathoki	9842154326 A5	
2	C2	Ram Rawal	9842154437 A7	
3	C3	Sabin Mukhia	9842154548 A9	
4	C4	Dilip Pun	9842154659 A3	
5	C5	Kabi Darlam	9842154770 A4	
6	C6	Ujwal Risal	9842154881 A9	
7	C7	Samyak Sakhya	9842154992 A1	
8	C8	Sumit Subba	9842155103 A2	
9	C9	Raju Shrestha	9842155214 A8	
10	C10	Dipesh Bhandari	9842155325 A3	

Figure 55 All data of **customer** table is displayed.

The screenshot shows the Oracle SQL Developer interface. In the top query editor window, the SQL command `select * from restaurant_dish;` is entered. Below it, the results are displayed in a table titled "Query Result". The table has two columns: "RESTAURANT_ID" and "DISH_CODE". The data consists of 15 rows, each containing a value for both columns. The "RESTAURANT_ID" column contains values 1 through 15, and the "DISH_CODE" column contains various dish codes such as CHPT, CKNUG, VGSUP, CKMO, T, etc.

RESTAURANT_ID	DISH_CODE	
1	R1	CHPT
2	R1	CKNUG
3	R1	VGSUP
4	R10	CKMO
5	R2	T
6	R3	CHPT
7	R5	CHPT
8	R5	CKMO
9	R6	CKNUG
10	R6	CKPZA
11	R7	T
12	R7	VGPZA
13	R8	VGSUP
14	R9	CKMO
15	R9	VGSUP

Figure 56 All data of **restaurant_dish** table is displayed.

The screenshot shows the Oracle SQL Developer interface. In the top-left query editor, the SQL command `select * from "Order";` is entered. Below the editor, the tabs `Script Output`, `Autotrace`, and `Query Result` are visible, with `Query Result` being the active tab. A status message indicates `All Rows Fetched: 15 in 0.009 seconds`. The results are displayed in a grid table with the following columns: ORDER_NO, TIME, ORDER_AMOUNT, STATUS, SN, and CUSTOMER_ID. The data rows are as follows:

	ORDER_NO	TIME	ORDER_AMOUNT	STATUS	SN	CUSTOMER_ID
1	O-1	1:01	1100	late	21-01	C4
2	O-2	2:22	1560	on time	21-08	C5
3	O-3	3:43	2020	on time	21-09	C6
4	O-4	5:04	2480	on time	21-05	C7
5	O-5	6:25	2940	early	21-06	C8
6	O-6	7:46	3400	early	21-02	C9
7	O-7	9:07	3860	late	21-07	C2
8	O-8	10:28	4320	on time	21-08	C3
9	O-9	11:49	4780	on time	21-09	C4
10	O-10	17:13	5240	late	21-05	C7
11	O-11	18:34	5700	on time	21-06	C8
12	O-12	5:04	6160	on time	21-02	C9
13	O-13	5:55	6620	on time	21-03	C2
14	O-14	4:55	7080	early	21-04	C3
15	O-15	5:32	7540	early	21-05	C4

Figure 57 All data of `order` table is displayed.

The screenshot shows the Oracle SQL Developer interface. In the top-left pane, a SQL editor window contains the query: `select * from dish_order;`. Below the editor, the results are displayed in a tabular format under the 'Query Result' tab. The results show 15 rows of data from the `dish_order` table, with columns labeled: DISH_CODE, ORDER_NO, ORDER_UNIT, LINE_TOTAL, and RESTAURANT_ID. The data includes various dish codes like VGSUP, CKMO, CHPT, etc., and order numbers ranging from 0-3 to 0-15. The total line amounts range from 60 to 300, and the restaurant IDs include R9, R5, R7, R1, R3, R6, R2, and R8.

	DISH_CODE	ORDER_NO	ORDER_UNIT	LINE_TOTAL	RESTAURANT_ID
1	VGSUP	0-3	1 plate	100	R9
2	CKMO	0-6	2 plate	200	R5
3	CHPT	0-7	3 plate	150	R5
4	T	0-8	2 plate	60	R7
5	CKNUG	0-9	1 plate	250	R1
6	VGSUP	0-10	3 plate	300	R1
7	CKMO	0-11	1 plate	100	R9
8	CHPT	0-12	2 plate	100	R3
9	CKPZA	0-13	1 plate	300	R6
10	VGPZA	0-2	1 plate	220	R7
11	CKMO	0-4	2 plate	200	R10
12	CHPT	0-5	2 plate	100	R1
13	T	0-6	2 plate	60	R2
14	CKNUG	0-7	1 plate	250	R6
15	VGSUP	0-4	1 plate	100	R8

Figure 58 All data of `dish_order` table is displayed.

13. User Manual

Create user manual showing all the unique steps associate with your Project.

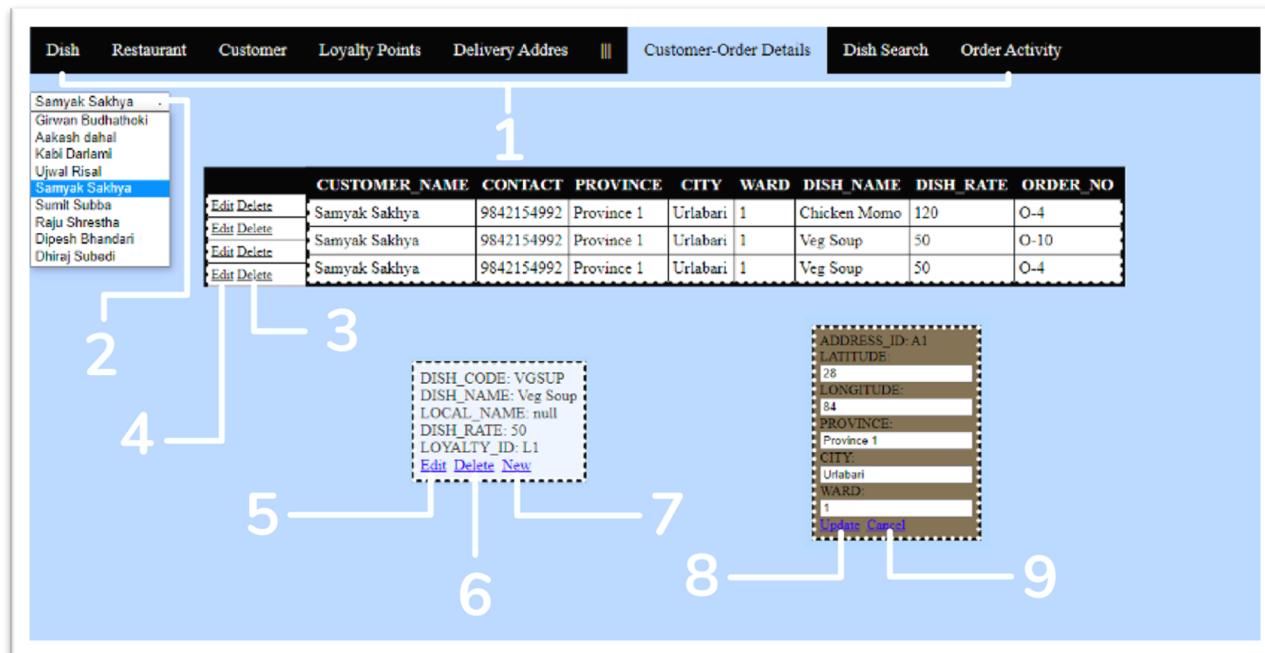


Figure 59 User Guide

1. This field is called navbar and it contains tabs that link to various pages of the application.
2. This button is called drop down button and it contains various data where you can select suitable button to get the needed data.
3. This is a delete button which gives you access to delete the entire row.
4. This is an edit button which gives you access to edit the attribute values.
5. This is an edit button which gives you access to edit the selected row.

6. This is a delete button which gives you access to delete the selected row.
7. This is an insert button which gives you access to insert values.
8. This is a update button which gives you access to update the inserted values.
9. This is a cancel button which cancels the ongoing insertion on the values.

14. Testing

Testing 1: Editing values in Dish form.

Dish Restaurant Customer Loyalty Points Delivery Address ||| Customer-Order Details Dish Search Order Activity

	DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID
Edit Delete	VGSUP	Veg Soup	null	50	L1
Edit Delete	CKMO	Chicken Momo	Chicken Momo	100	L1
Edit Delete	CHPT	Chapatti	roti	50	L2
Edit Delete	T	Tea	chiya	30	L4
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7
Edit Delete	VGBGR	Veg Burger	null	80	L8
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2
Edit Delete	CKPZA	Chicken Pizza	null	300	L10
Edit Delete	VGPZA	Veg Pizza	null	220	L3

Figure 60 Before editing highlighted values in Dish form.

The screenshot shows a table of dish data with columns: DISH_CODE, DISH_NAME, LOCAL_NAME, DISH_RATE, and LOYALTY_ID. A tooltip box is overlaid on the first row, containing the following information:

```

DISH_CODE: VGSUP
DISH_NAME: Veg Soup
LOCAL_NAME: null
DISH_RATE: 50
LOYALTY_ID: L1
Edit Delete New

```

	DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID
Edit Delete	VGSUP	Veg Soup	null	50	L1
Update Cancel	CKMO	Chicken Momo	Kukhura momo	120	L2
Edit Delete	CHPT	Chapatti	roti	50	L2
Edit Delete	T	Tea	chiya	30	L4
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7
Edit Delete	VGBGR	Veg Burger	null	80	L8
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2
Edit Delete	CKPZA	Chicken Pizza	null	300	L10
Edit Delete	VGPZA	Veg Pizza	null	220	L3

Figure 61 while editing highlighted values in Dish form.

The screenshot shows the same table of dish data. The first row has been updated: DISH_CODE is now VGSUP, DISH_NAME is Veg Soup, LOCAL_NAME is null, DISH_RATE is 50, and LOYALTY_ID is L1. A tooltip box is overlaid on the first row, containing the following information:

```

DISH_CODE: VGSUP
DISH_NAME: Veg Soup
LOCAL_NAME: null
DISH_RATE: 50
LOYALTY_ID: L1
Edit Delete New

```

	DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID
Edit Delete	VGSUP	Veg Soup	null	50	L1
Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2
Edit Delete	CHPT	Chapatti	roti	50	L2
Edit Delete	T	Tea	chiya	30	L4
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7
Edit Delete	VGBGR	Veg Burger	null	80	L8
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2
Edit Delete	CKPZA	Chicken Pizza	null	300	L10
Edit Delete	VGPZA	Veg Pizza	null	220	L3

Figure 62 After editing highlighted values in Dish form.

Testing 2: Deleting values in dish form.

Dish	Restaurant	Customer	Loyalty Points	Delivery Address		Customer-Order Details	Dish Search	Order Activity																																																																	
<table border="1"> <thead> <tr> <th>DISH_CODE</th><th>DISH_NAME</th><th>LOCAL_NAME</th><th>DISH_RATE</th><th>LOYALTY_ID</th></tr> </thead> <tbody> <tr><td>Edit Delete</td><td>VGSUP</td><td>Veg Soup</td><td>null</td><td>50</td><td>L1</td></tr> <tr><td>Edit Delete</td><td>CKMO</td><td>Chicken Momo</td><td>Kukhura momo</td><td>120</td><td>L2</td></tr> <tr><td>Edit Delete</td><td>CHPT</td><td>Chapatti</td><td>roti</td><td>50</td><td>L2</td></tr> <tr><td>Edit Delete</td><td>T</td><td>Tea</td><td>chiya</td><td>30</td><td>L4</td></tr> <tr><td>Edit Delete</td><td>CKNUG</td><td>Chicken Nuggets</td><td>null</td><td>250</td><td>L4</td></tr> <tr><td>Edit Delete</td><td>CKSEK</td><td>Chicken Sekuwa</td><td>null</td><td>200</td><td>L7</td></tr> <tr><td>Edit Delete</td><td>VGBGR</td><td>Veg Burger</td><td>null</td><td>80</td><td>L8</td></tr> <tr><td>Edit Delete</td><td>BFSKT</td><td>Buff Sukuti</td><td>buff sukuti</td><td>250</td><td>L2</td></tr> <tr><td>Edit Delete</td><td>CKPZA</td><td>Chicken Pizza</td><td>null</td><td>300</td><td>L10</td></tr> <tr><td>Edit Delete</td><td>VGPZA</td><td>Veg Pizza</td><td>null</td><td>220</td><td>L3</td></tr> </tbody> </table>									DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID	Edit Delete	VGSUP	Veg Soup	null	50	L1	Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2	Edit Delete	CHPT	Chapatti	roti	50	L2	Edit Delete	T	Tea	chiya	30	L4	Edit Delete	CKNUG	Chicken Nuggets	null	250	L4	Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7	Edit Delete	VGBGR	Veg Burger	null	80	L8	Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2	Edit Delete	CKPZA	Chicken Pizza	null	300	L10	Edit Delete	VGPZA	Veg Pizza	null	220	L3
DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID																																																																					
Edit Delete	VGSUP	Veg Soup	null	50	L1																																																																				
Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2																																																																				
Edit Delete	CHPT	Chapatti	roti	50	L2																																																																				
Edit Delete	T	Tea	chiya	30	L4																																																																				
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4																																																																				
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7																																																																				
Edit Delete	VGBGR	Veg Burger	null	80	L8																																																																				
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2																																																																				
Edit Delete	CKPZA	Chicken Pizza	null	300	L10																																																																				
Edit Delete	VGPZA	Veg Pizza	null	220	L3																																																																				

Figure 63 Before deleting highlighted values in Dish form.

Dish	Restaurant	Customer	Loyalty Points	Delivery Address		Customer-Order Details	Dish Search	Order Activity																																																											
<table border="1"> <thead> <tr> <th>DISH_CODE</th><th>DISH_NAME</th><th>LOCAL_NAME</th><th>DISH_RATE</th><th>LOYALTY_ID</th></tr> </thead> <tbody> <tr><td>Edit Delete</td><td>VGSUP</td><td>Veg Soup</td><td>null</td><td>50</td><td>L1</td></tr> <tr><td>Edit Delete</td><td>CKMO</td><td>Chicken Momo</td><td>Kukhura momo</td><td>120</td><td>L2</td></tr> <tr><td>Edit Delete</td><td>CHPT</td><td>Chapatti</td><td>roti</td><td>50</td><td>L2</td></tr> <tr><td>Edit Delete</td><td>CKNUG</td><td>Chicken Nuggets</td><td>null</td><td>250</td><td>L4</td></tr> <tr><td>Edit Delete</td><td>CKSEK</td><td>Chicken Sekuwa</td><td>null</td><td>200</td><td>L7</td></tr> <tr><td>Edit Delete</td><td>VGBGR</td><td>Veg Burger</td><td>null</td><td>80</td><td>L8</td></tr> <tr><td>Edit Delete</td><td>BFSKT</td><td>Buff Sukuti</td><td>buff sukuti</td><td>250</td><td>L2</td></tr> <tr><td>Edit Delete</td><td>CKPZA</td><td>Chicken Pizza</td><td>null</td><td>300</td><td>L10</td></tr> <tr><td>Edit Delete</td><td>VGPZA</td><td>Veg Pizza</td><td>null</td><td>220</td><td>L3</td></tr> </tbody> </table>									DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID	Edit Delete	VGSUP	Veg Soup	null	50	L1	Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2	Edit Delete	CHPT	Chapatti	roti	50	L2	Edit Delete	CKNUG	Chicken Nuggets	null	250	L4	Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7	Edit Delete	VGBGR	Veg Burger	null	80	L8	Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2	Edit Delete	CKPZA	Chicken Pizza	null	300	L10	Edit Delete	VGPZA	Veg Pizza	null	220	L3
DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID																																																															
Edit Delete	VGSUP	Veg Soup	null	50	L1																																																														
Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2																																																														
Edit Delete	CHPT	Chapatti	roti	50	L2																																																														
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4																																																														
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7																																																														
Edit Delete	VGBGR	Veg Burger	null	80	L8																																																														
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2																																																														
Edit Delete	CKPZA	Chicken Pizza	null	300	L10																																																														
Edit Delete	VGPZA	Veg Pizza	null	220	L3																																																														

Figure 64 After deleting highlighted values in Dish form.

Testing 3: Inserting values in dish form.

Dish Restaurant Customer Loyalty Points Delivery Address ||| Customer-Order Details Dish Search Order Activity

	DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID
Edit Delete	VGSUP	Veg Soup	null	50	L1
Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2
Edit Delete	CHPT	Chapatti	roti	50	L2
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7
Edit Delete	VGBGR	Veg Burger	null	80	L8
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2
Edit Delete	CKPZA	Chicken Pizza	null	300	L10
Edit Delete	VGPZA	Veg Pizza	null	220	L3

DISH_CODE: VGSUP
 DISH_NAME: Veg Soup
 LOCAL_NAME: null
 DISH_RATE: 50
 LOYALTY_ID: L1
[Edit](#) [Delete](#) [New](#)

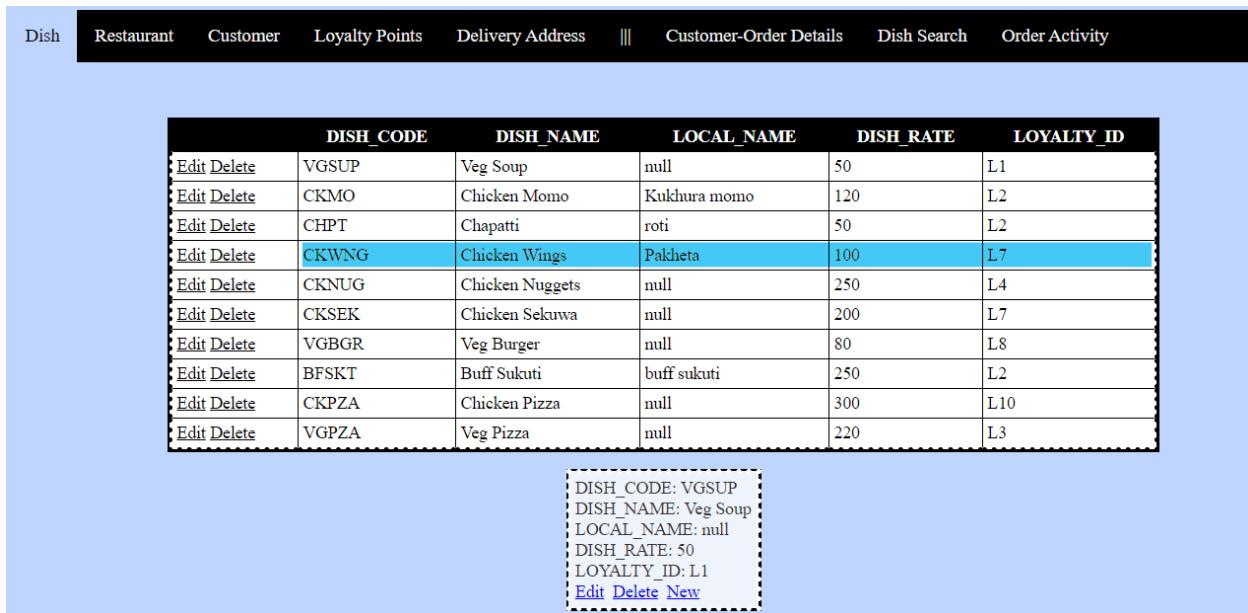
Figure 65 Before inserting values in Dish form.

Dish Restaurant Customer Loyalty Points Delivery Address ||| Customer-Order Details Dish Search Order Activity

	DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID
Edit Delete	VGSUP	Veg Soup	null	50	L1
Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2
Edit Delete	CHPT	Chapatti	roti	50	L2
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7
Edit Delete	VGBGR	Veg Burger	null	80	L8
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2
Edit Delete	CKPZA	Chicken Pizza	null	300	L10
Edit Delete	VGPZA	Veg Pizza	null	220	L3

DISH_CODE: CKWNG
 DISH_NAME: Chicken Wings
 LOCAL_NAME: Pakheta
 DISH_RATE: 100
 LOYALTY_ID: L7
[Insert](#) [Cancel](#)

Figure 66 While inserting highlighted values in Dish form.



The screenshot shows a web-based application interface for managing dish records. At the top, there is a navigation bar with links: Dish, Restaurant, Customer, Loyalty Points, Delivery Address, |||, Customer-Order Details, Dish Search, and Order Activity. Below the navigation bar is a table with the following columns: DISH_CODE, DISH_NAME, LOCAL_NAME, DISH_RATE, and LOYALTY_ID. The table contains ten rows of data. The fifth row, which corresponds to the highlighted row in the screenshot, has the following values:

	DISH_CODE	DISH_NAME	LOCAL_NAME	DISH_RATE	LOYALTY_ID
Edit Delete	VGSUP	Veg Soup	null	50	L1
Edit Delete	CKMO	Chicken Momo	Kukhura momo	120	L2
Edit Delete	CHPT	Chapatti	roti	50	L2
Edit Delete	CKWNG	Chicken Wings	Pakheta	100	L7
Edit Delete	CKNUG	Chicken Nuggets	null	250	L4
Edit Delete	CKSEK	Chicken Sekuwa	null	200	L7
Edit Delete	VGBGR	Veg Burger	null	80	L8
Edit Delete	BFSKT	Buff Sukuti	buff sukuti	250	L2
Edit Delete	CKPZA	Chicken Pizza	null	300	L10
Edit Delete	VGPZA	Veg Pizza	null	220	L3

A tooltip box is overlaid on the fifth row, containing the following information:

DISH_CODE: VGSUP
DISH_NAME: Veg Soup
LOCAL_NAME: null
DISH_RATE: 50
LOYALTY_ID: L1
[Edit](#) [Delete](#) [New](#)

Figure 67 After inserting highlighted values in Dish form.

Testing 4: Editing values in restaurant form.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Edit Delete	R3	Nepali Restaurant	555333
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R9	Sigma	555999
Edit Delete	R10	Boudha Restro	555110

RESTAURANT_ID: R1
 AVAILABLE_ON_RESTAURANT: Restaurant Nepal
 TELEPHONE_NO: 555111
[Edit](#) [Delete](#) [New](#)

Figure 68 Before editing highlighted values in restaurant form.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Update Cancel	R3	Eats	541777
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R9	Sigma	555999
Edit Delete	R10	Boudha Restro	555110

RESTAURANT_ID: R1
 AVAILABLE_ON_RESTAURANT: Restaurant Nepal

Figure 69 While editing highlighted values in restaurant form.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Edit Delete	R3	Eats	541777
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R9	Sigma	555999
Edit Delete	R10	Boudha Restro	555110

RESTAURANT_ID: R1
AVAILABLE ON RESTAURANT: Restaurant

Figure 70 After editing highlighted values in restaurant form.

Testing 5: Deleting in restaurant.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit	R1	Restaurant Nepal	555111
Edit	R2	Kitchen Urlabari	555222
Edit	R3	Eats	541777
Edit	R4	Test Of Nepal	555444
Edit	R5	Momo Vibe	555555
Edit	R6	Nepalese Flavor	555666
Edit	R7	Grill Center	555777
Edit	R8	Sekuwa Corner	555888
Edit	R9	Sigma	555999
Edit	R10	Boudha Restro	555110

RESTAURANT_ID: R1
AVAILABLE_ON_RESTAURANT: Restaurant

Figure 71 Before deleting highlighted values in restaurant form.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit	R1	Restaurant Nepal	555111
Edit	R2	Kitchen Urlabari	555222
Edit	R3	Eats	541777
Edit	R4	Test Of Nepal	555444
Edit	R5	Momo Vibe	555555
Edit	R6	Nepalese Flavor	555666
Edit	R7	Grill Center	555777
Edit	R8	Sekuwa Corner	555888
Edit	R10	Boudha Restro	555110

RESTAURANT_ID: R1
AVAILABLE_ON_RESTAURANT: Restaurant
Nepal

Figure 72 After deleting highlighted values in restaurant form.

Testing 6: Inserting in restaurant.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Edit Delete	R3	Eats	541777
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R10	Boudha Restro	555110

RESTAURANT_ID: R1
AVAILABLE_ON_RESTAURANT: Restaurant Nepal

Figure 73 Before inserting values in restaurant form.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Edit Delete	R3	Eats	541777
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R10	Boudha Restro	555110

RESTAURANT_ID: R9
AVAILABLE_ON_RESTAURANT:
Totala
TELEPHONE_NO: 544554
[Insert](#) [Cancel](#)

Figure 74 While inserting highlighted values in restaurant form.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Edit Delete	R3	Eats	541777
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R10	Boudha Restro	555110
Edit Delete	R9	Totala	544554

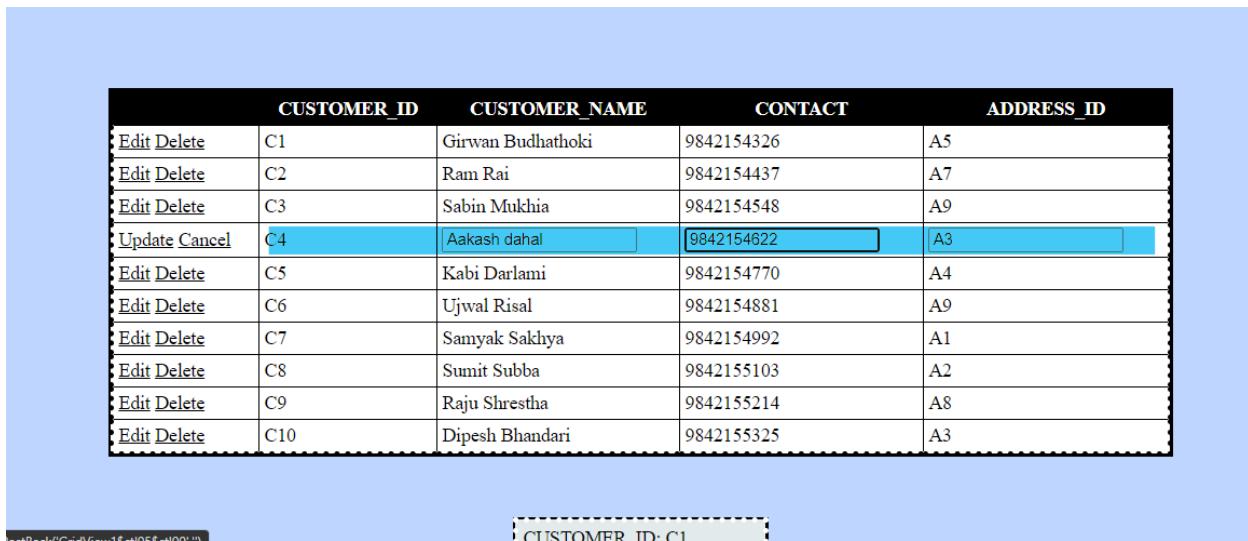
RESTAURANT_ID: R1
AVAILABLE_ON_RESTAURANT: Restaurant

Figure 75 After inserting highlighted values in restaurant form.

Testing 7: Edit in customer form.


	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit Delete	C1	Girwan Budhathoki	9842154326	A5
Edit Delete	C2	Ram Rai	9842154437	A7
Edit Delete	C3	Sabin Mukhia	9842154548	A9
Edit Delete	C4	Dilip Pun	9842154659	A3
Edit Delete	C5	Kabi Darlami	9842154770	A4
Edit Delete	C6	Ujwal Risal	9842154881	A9
Edit Delete	C7	Samyak Sakhya	9842154992	A1
Edit Delete	C8	Sumit Subba	9842155103	A2
Edit Delete	C9	Raju Shrestha	9842155214	A8
Edit Delete	C10	Dipesh Bhandari	9842155325	A3

Figure 76 Before editing highlighted values in customer form.



	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit Delete	C1	Girwan Budhathoki	9842154326	A5
Edit Delete	C2	Ram Rai	9842154437	A7
Edit Delete	C3	Sabin Mukhia	9842154548	A9
Update Cancel	C4	Aakash dahal	9842154622	A3
Edit Delete	C5	Kabi Darlami	9842154770	A4
Edit Delete	C6	Ujwal Risal	9842154881	A9
Edit Delete	C7	Samyak Sakhya	9842154992	A1
Edit Delete	C8	Sumit Subba	9842155103	A2
Edit Delete	C9	Raju Shrestha	9842155214	A8
Edit Delete	C10	Dipesh Bhandari	9842155325	A3

Figure 77 While editing highlighted values in customer form.

	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit Delete	C1	Girwan Budhathoki	9842154326	A5
Edit Delete	C2	Ram Rai	9842154437	A7
Edit Delete	C3	Sabin Mukhia	9842154548	A9
Edit Delete	C4	Aakash dahal	9842154622	A3
Edit Delete	C5	Kabi Darlami	9842154770	A4
Edit Delete	C6	Ujwal Risal	9842154881	A9
Edit Delete	C7	Samyak Sakhya	9842154992	A1
Edit Delete	C8	Sumit Subba	9842155103	A2
Edit Delete	C9	Raju Shrestha	9842155214	A8
Edit Delete	C10	Dipesh Bhandari	9842155325	A3

CUSTOMER_ID: C1

Figure 78 After editing highlighted values in customer form.

Testing 8: Deleting in customer form.

	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit	Delete	Girwan Budhathoki	9842154326	A5
Edit	Delete	Ram Rai	9842154437	A7
Edit	Delete	C3	Sabin Mukhia	9842154548
Edit	Delete	Aakash dahal	9842154622	A3
Edit	Delete	Kabi Darlami	9842154770	A4
Edit	Delete	Ujwal Risal	9842154881	A9
Edit	Delete	Samyak Sakhya	9842154992	A1
Edit	Delete	Sumit Subba	9842155103	A2
Edit	Delete	Raju Shrestha	9842155214	A8
Edit	Delete	Dipesh Bhandari	9842155325	A3

CUSTOMER_ID: C1
CUSTOMER_NAME: Girwan

Figure 79 Before deleting highlighted values in customer form.

	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit	Delete	Girwan Budhathoki	9842154326	A5
Edit	Delete	Ram Rai	9842154437	A7
Edit	Delete	Aakash dahal	9842154622	A3
Edit	Delete	Kabi Darlami	9842154770	A4
Edit	Delete	Ujwal Risal	9842154881	A9
Edit	Delete	Samyak Sakhya	9842154992	A1
Edit	Delete	Sumit Subba	9842155103	A2
Edit	Delete	Raju Shrestha	9842155214	A8
Edit	Delete	Dipesh Bhandari	9842155325	A3

CUSTOMER_ID: C1
CUSTOMER_NAME: Girwan

Figure 80 After deleting highlighted values in customer form.

Testing 9: Inserting in customer form.

	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit	C1	Girwan Budhathoki	9842154326	A5
Edit	C2	Ram Rai	9842154437	A7
Edit	C4	Aakash dahal	9842154622	A3
Edit	C5	Kabi Darlami	9842154770	A4
Edit	C6	Ujwal Risal	9842154881	A9
Edit	C7	Samyak Sakhya	9842154992	A1
Edit	C8	Sumit Subba	9842155103	A2
Edit	C9	Raju Shrestha	9842155214	A8
Edit	C10	Dipesh Bhandari	9842155325	A3

CUSTOMER_ID: C1
CUSTOMER_NAME: Girwan

Figure 81 Before inserting highlighted values in customer form.

	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit	C1	Girwan Budhathoki	9842154326	A5
Edit	C2	Ram Rai	9842154437	A7
Edit	C4	Aakash dahal	9842154622	A3
Edit	C5	Kabi Darlami	9842154770	A4
Edit	C6	Ujwal Risal	9842154881	A9
Edit	C7	Samyak Sakhya	9842154992	A1
Edit	C8	Sumit Subba	9842155103	A2
Edit	C9	Raju Shrestha	9842155214	A8
Edit	C10	Dipesh Bhandari	9842155325	A3

CUSTOMER_ID:
C11
CUSTOMER_NAME:
Dhiraj Subedi
CONTACT:
9842121498
ADDRESS_ID:
A1
[Insert](#) [Cancel](#)

Figure 82 while inserting highlighted values in customer form.

	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit Delete	C1	Girwan Budhathoki	9842154326	A5
Edit Delete	C2	Ram Rai	9842154437	A7
Edit Delete	C4	Aakash dahal	9842154622	A3
Edit Delete	C5	Kabi Darlami	9842154770	A4
Edit Delete	C6	Ujwal Risal	9842154881	A9
Edit Delete	C7	Samyak Sakhya	9842154992	A1
Edit Delete	C8	Sumit Subba	9842155103	A2
Edit Delete	C9	Raju Shrestha	9842155214	A8
Edit Delete	C10	Dipesh Bhandari	9842155325	A3
Edit Delete	C11	Dhiraj Subedi	9842121498	A1

CUSTOMER_ID: C1
CUSTOMER_NAME: Girwan Budhathoki
CONTACT: 9842154326
ADDRESS_ID: A5
[Edit](#) [Delete](#) [New](#)

Figure 83 After inserting highlighted values in customer form.

Testing 10: Editing values in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5
Edit Delete	L11	Saturday	10

LOYALTY_ID: L1
 DAY: Wednesday
 LOYALTY_POINT: 10
[Edit](#) [Delete](#) [New](#)

Figure 84 Before editing highlighted values in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5
Update Cancel	L11	Monday	12

LOYALTY_ID: L1
 DAY: Wednesday
 LOYALTY_POINT: 10
[Edit](#) [Delete](#) [New](#)

Figure 85 While editing highlighted values in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5
Edit Delete	L11	Monday	12

LOYALTY_ID: L1
DAY: Wednesday
LOYALTY_POINT: 10
[Edit](#) [Delete](#) [New](#)

Figure 86 After editing highlighted values in loyalty points form.

Testing 11: Deleting in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5
Edit Delete	L11	Monday	12

LOYALTY_ID: L1
DAY: Wednesday
LOYALTY_POINT: 10
[Edit](#) [Delete](#) [New](#)

Figure 87 Before deleting highlighted values in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5

LOYALTY_ID: L1
DAY: Wednesday
LOYALTY_POINT: 10
[Edit](#) [Delete](#) [New](#)

Figure 88 After deleting highlighted values in loyalty points form.

Testing 12: Inserting values in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5

LOYALTY_ID: L1
 DAY: Wednesday
 LOYALTY_POINT: 10
[Edit](#) [Delete](#) [New](#)

Figure 89 Before inserting values in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5

LOYALTY_ID:

 DAY:

 LOYALTY POINT:

[Insert](#) [Cancel](#)

Figure 90 While inserting highlighted values in loyalty points form.

	LOYALTY_ID	DAY	LOYALTY_POINT
Edit Delete	L1	Wednesday	10
Edit Delete	L2	Thursday	5
Edit Delete	L3	Friday	15
Edit Delete	L4	Saturday	20
Edit Delete	L5	Sunday	10
Edit Delete	L6	Monday	12
Edit Delete	L7	Tuesday	5
Edit Delete	L8	Wednesday	15
Edit Delete	L9	Thursday	20
Edit Delete	L10	Friday	5

LOYALTY_ID: L1
DAY: Wednesday
LOYALTY_POINT: 10
[Edit](#) [Delete](#) [New](#)

Figure 91 After inserting highlighted values in loyalty points form.

Testing 13: Editing values in Delivery address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit Delete	A1	28	84	Province 1	Urlabari	1
Edit Delete	A2	29	85	Province 1	Urlabari	2
Edit Delete	A3	30	86	Province 1	Urlabari	3
Edit Delete	A4	31	87	Province 1	Urlabari	4
Edit Delete	A5	32	88	Province 1	Urlabari	5
Edit Delete	A6	33	89	Province 1	Damak	6
Edit Delete	A7	34	90	Province 1	Damak	7
Edit Delete	A8	35	91	Province 1	Damak	8
Edit Delete	A9	36	92	Province 1	Damak	9
Edit Delete	A10	37	92	Province 1	Damak	10

ADDRESS_ID: A1
 LATITUDE: 28
 LONGITUDE: 84
 PROVINCE: Province 1
 CITY: Urlabari
 WARD: 1
[Edit](#) [Delete](#) [New](#)

Figure 92 Before editing highlighted values in Delivery address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit Delete	A1	28	84	Province 1	Urlabari	1
Edit Delete	A2	29	85	Province 1	Urlabari	2
Edit Delete	A3	30	86	Province 1	Urlabari	3
Edit Delete	A4	31	87	Province 1	Urlabari	4
Edit Delete	A5	32	88	Province 1	Urlabari	5
Update Cancel	A6	29	84	Province 1	Urlabari	6
Edit Delete	A7	34	90	Province 1	Damak	7
Edit Delete	A8	35	91	Province 1	Damak	8
Edit Delete	A9	36	92	Province 1	Damak	9
Edit Delete	A10	37	92	Province 1	Damak	10

ADDRESS_ID: A1
 LATITUDE: 28
 LONGITUDE: 84
 PROVINCE: Province 1
 CITY: Urlabari
 WARD: 1
[Edit](#) [Delete](#) [New](#)

Figure 93 While editing highlighted values in Delivery address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit Delete	A1	28	84	Province 1	Urlabari	1
Edit Delete	A2	29	85	Province 1	Urlabari	2
Edit Delete	A3	30	86	Province 1	Urlabari	3
Edit Delete	A4	31	87	Province 1	Urlabari	4
Edit Delete	A5	32	88	Province 1	Urlabari	5
Edit Delete	A6	29	84	Province 1	Urlabari	6
Edit Delete	A7	34	90	Province 1	Damak	7
Edit Delete	A8	35	91	Province 1	Damak	8
Edit Delete	A9	36	92	Province 1	Damak	9
Edit Delete	A10	37	92	Province 1	Damak	10

ADDRESS_ID: A1
 LATITUDE: 28
 LONGITUDE: 84
 PROVINCE: Province 1
 CITY: Urlabari
 WARD: 1
[Edit](#) [Delete](#) [New](#)

Figure 94 After editing highlighted values in Delivery address form.

Testing 14: Deleting in Delivery Address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit	A1	28	84	Province 1	Urlabari	1
Edit	A2	29	85	Province 1	Urlabari	2
Edit	A3	30	86	Province 1	Urlabari	3
Edit	A4	31	87	Province 1	Urlabari	4
Edit	A5	32	88	Province 1	Urlabari	5
Edit	A6	29	84	Province 1	Urlabari	6
Edit	A7	34	90	Province 1	Damak	7
Edit	A8	35	91	Province 1	Damak	8
Edit	A9	36	92	Province 1	Damak	9
Edit	A10	37	92	Province 1	Damak	10

ADDRESS_ID: A1
 LATITUDE: 28
 LONGITUDE: 84
 PROVINCE: Province 1
 CITY: Urlabari
 WARD: 1
[Edit](#) [Delete](#) [New](#)

Figure 95 Before deleting highlighted values in Delivery address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit	A1	28	84	Province 1	Urlabari	1
Edit	A2	29	85	Province 1	Urlabari	2
Edit	A3	30	86	Province 1	Urlabari	3
Edit	A4	31	87	Province 1	Urlabari	4
Edit	A5	32	88	Province 1	Urlabari	5
Edit	A6	29	84	Province 1	Urlabari	6
Edit	A8	35	91	Province 1	Damak	8
Edit	A9	36	92	Province 1	Damak	9
Edit	A10	37	92	Province 1	Damak	10

ADDRESS_ID: A1
 LATITUDE: 28
 LONGITUDE: 84
 PROVINCE: Province 1
 CITY: Urlabari

Figure 96 After deleting highlighted values in Delivery address form.

Testing 15: Inserting in Delivery address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit Delete	A1	28	84	Province 1	Urlabari	1
Edit Delete	A2	29	85	Province 1	Urlabari	2
Edit Delete	A3	30	86	Province 1	Urlabari	3
Edit Delete	A4	31	87	Province 1	Urlabari	4
Edit Delete	A5	32	88	Province 1	Urlabari	5
Edit Delete	A6	29	84	Province 1	Urlabari	6
Edit Delete	A8	35	91	Province 1	Damak	8
Edit Delete	A9	36	92	Province 1	Damak	9
Edit Delete	A10	37	92	Province 1	Damak	10

ADDRESS_ID: A1
LATITUDE: 28
LONGITUDE: 84
PROVINCE: Province 1
CITY: Urlabari

Figure 97 Before inserting values in Delivery address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit Delete	A1	28	84	Province 1	Urlabari	1
Edit Delete	A2	29	85	Province 1	Urlabari	2
Edit Delete	A3	30	86	Province 1	Urlabari	3
Edit Delete	A4	31	87	Province 1	Urlabari	4
Edit Delete	A5	32	88	Province 1	Urlabari	5
Edit Delete	A6	29	84	Province 1	Urlabari	6
Edit Delete	A8	35	91	Province 1	Damak	8
Edit Delete	A9	36	92	Province 1	Damak	9
Edit Delete	A10	37	92	Province 1	Damak	10

ADDRESS_ID:
A7

LATITUDE:
33

LONGITUDE:
88

PROVINCE:
Province 1

CITY:
Damak

WARD:
7

[Insert](#) [Cancel](#)

Figure 98 While inserting highlighted values in Delivery address form.

	ADDRESS_ID	LATITUDE	LONGITUDE	PROVINCE	CITY	WARD
Edit	A1	28	84	Province 1	Urlabari	1
Edit	A2	29	85	Province 1	Urlabari	2
Edit	A3	30	86	Province 1	Urlabari	3
Edit	A4	31	87	Province 1	Urlabari	4
Edit	A5	32	88	Province 1	Urlabari	5
Edit	A6	29	84	Province 1	Urlabari	6
Edit	A7	33	88	Province 1	Damak	7
Edit	A8	35	91	Province 1	Damak	8
Edit	A9	36	92	Province 1	Damak	9
Edit	A10	37	92	Province 1	Damak	10

ADDRESS_ID: A1
 LATITUDE: 28
 LONGITUDE: 84
 PROVINCE: Province 1
 CITY: Urlabari
 WARD: 1
[Edit](#) [Delete](#) [New](#)

Figure 99 After inserting highlighted values in Delivery address form.

Testing 16: Data filtering on customer order details form.

The screenshot shows a web-based application interface. At the top, there is a navigation bar with links: Dish, Restaurant, Customer, Loyalty Points, Delivery Address, Customer-Order Details, Dish Search, and Order Activity. Below the navigation bar, a dropdown menu is open, showing the option 'Aakash dahal'. The main content area displays a table titled 'Customer-Order Details' with the following data:

CUSTOMER_NAME	CONTACT	PROVINCE	CITY	WARD	DISH_NAME	DISH_RATE	ORDER_NO
Aakash dahal	9842154622	Province 1	Urlabari	3	Chicken Nuggets	250	O-9

Figure 100 Displaying customer order detail of aakash dahal.

The screenshot shows a web-based application interface, similar to Figure 100. At the top, there is a navigation bar with links: Dish, Restaurant, Customer, Loyalty Points, Delivery Address, Customer-Order Details, Dish Search, and Order Activity. Below the navigation bar, a dropdown menu is open, showing the option 'Samyak Sakhya'. The main content area displays a table titled 'Customer-Order Details' with the following data:

CUSTOMER_NAME	CONTACT	PROVINCE	CITY	WARD	DISH_NAME	DISH_RATE	ORDER_NO
Samyak Sakhya	9842154992	Province 1	Urlabari	1	Chicken Momo	120	O-4
Samyak Sakhya	9842154992	Province 1	Urlabari	1	Veg Soup	50	O-10
Samyak Sakhya	9842154992	Province 1	Urlabari	1	Veg Soup	50	O-4

Figure 101 Displaying order details of Samyak Sakhya.

Testing 17: Data filtering on dish search form.

Dish	Restaurant	Customer	Loyalty Points	Delivery Address		Customer-Order Details	Dish Search	Order Activity																	
Veg Soup						<table border="1"><thead><tr><th>DISH_CODE</th><th>DISH_NAME</th><th>LOCAL_NAME</th><th>AVAILABLE_ON_RESTAURANT</th><th>TELEPHONE_NO</th></tr></thead><tbody><tr><td>VGSUP</td><td>Veg Soup</td><td>null</td><td>Restaurant Nepal</td><td>555111</td></tr><tr><td>VGSUP</td><td>Veg Soup</td><td>null</td><td>Sekuwa Corner</td><td>555888</td></tr></tbody></table>	DISH_CODE	DISH_NAME	LOCAL_NAME	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO	VGSUP	Veg Soup	null	Restaurant Nepal	555111	VGSUP	Veg Soup	null	Sekuwa Corner	555888				
DISH_CODE	DISH_NAME	LOCAL_NAME	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO																					
VGSUP	Veg Soup	null	Restaurant Nepal	555111																					
VGSUP	Veg Soup	null	Sekuwa Corner	555888																					

Figure 102 Displaying Veg Soup availability.

Chapatti		<table border="1"><thead><tr><th>DISH_CODE</th><th>DISH_NAME</th><th>LOCAL_NAME</th><th>AVAILABLE_ON_RESTAURANT</th><th>TELEPHONE_NO</th></tr></thead><tbody><tr><td>CHPT</td><td>Chapatti</td><td>roti</td><td>Restaurant Nepal</td><td>555111</td></tr><tr><td>CHPT</td><td>Chapatti</td><td>roti</td><td>Eats</td><td>541777</td></tr><tr><td>CHPT</td><td>Chapatti</td><td>roti</td><td>Momo Vibe</td><td>555555</td></tr></tbody></table>	DISH_CODE	DISH_NAME	LOCAL_NAME	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO	CHPT	Chapatti	roti	Restaurant Nepal	555111	CHPT	Chapatti	roti	Eats	541777	CHPT	Chapatti	roti	Momo Vibe	555555					
DISH_CODE	DISH_NAME	LOCAL_NAME	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO																							
CHPT	Chapatti	roti	Restaurant Nepal	555111																							
CHPT	Chapatti	roti	Eats	541777																							
CHPT	Chapatti	roti	Momo Vibe	555555																							

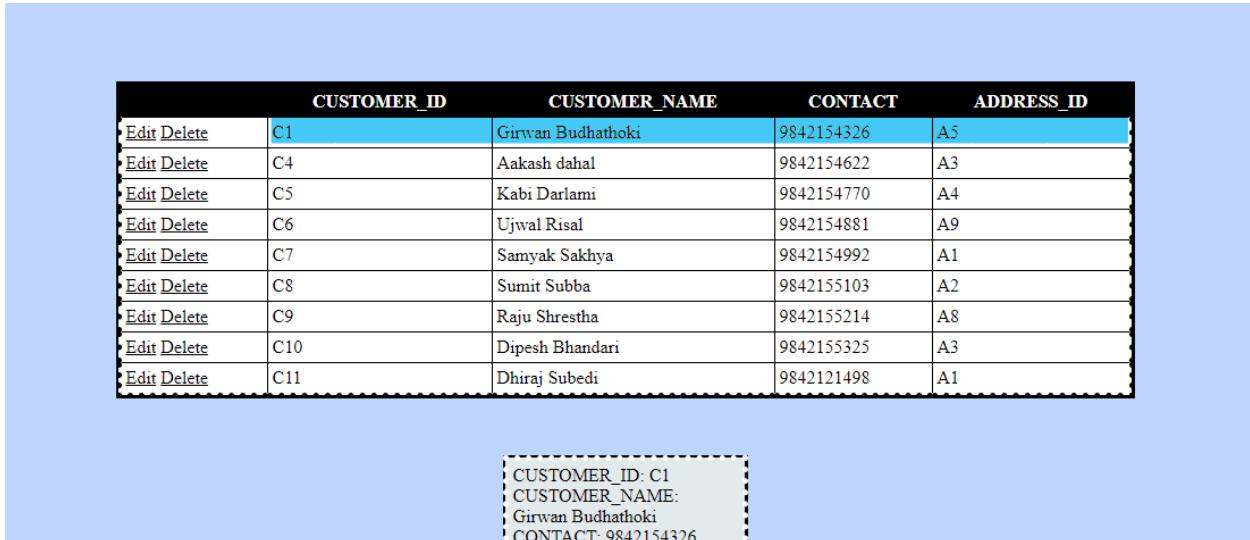
Figure 103 Displaying Chapatti availability.

Testing 18: Data filtering on Order Activity form.

Dish	Restaurant	Customer	Loyalty Points	Delivery Address		Customer-Order Details	Dish Search	Order Activity	
DEC ▾									
ORDER_NO RESTAURANT_ID AVAILABLE_ON_RESTAURANT TOTAL_ORDER Date									
O-5	R1	Restaurant Nepal	1	12/22/2021 12:00:00 AM					
O-6	R5	Momo Vibe	1	12/14/2021 12:00:00 AM					
O-9	R1	Restaurant Nepal	1	12/16/2021 12:00:00 AM					
O-4	R10	Boudha Restro	1	12/16/2021 12:00:00 AM					
O-4	R8	Sekuwa Corner	1	12/16/2021 12:00:00 AM					

Table 13 Order history is displayed.

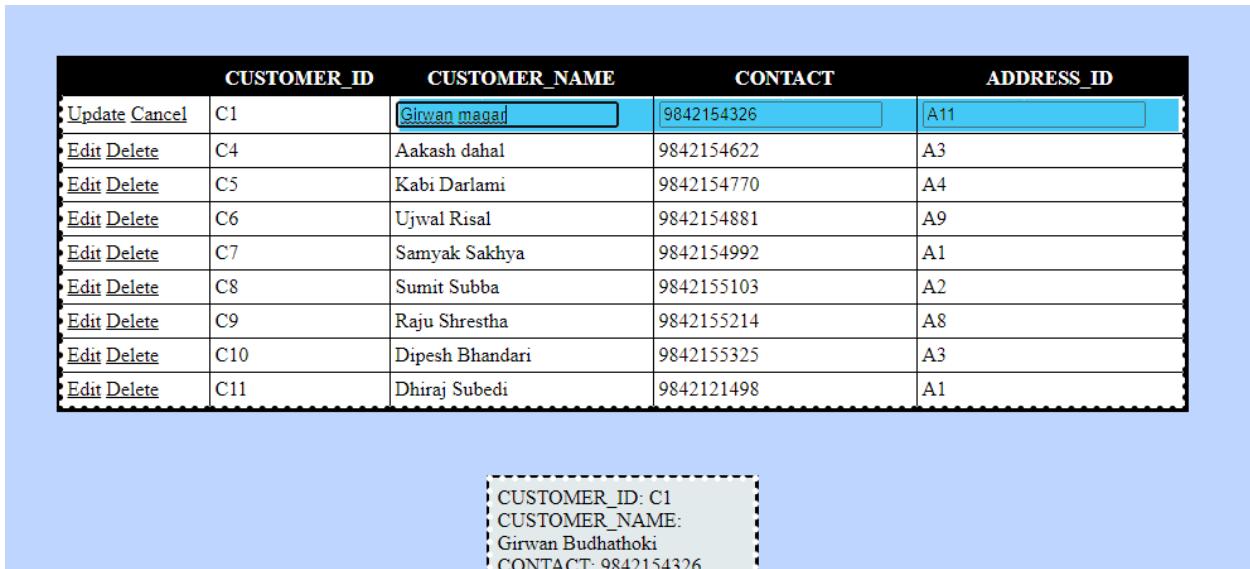
Testing 19: Foreign Key violation.



	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Edit Delete	C1	Girwan Budhathoki	9842154326	A5
Edit Delete	C4	Aakash dahal	9842154622	A3
Edit Delete	C5	Kabi Darlami	9842154770	A4
Edit Delete	C6	Ujwal Risal	9842154881	A9
Edit Delete	C7	Samyak Sakhya	9842154992	A1
Edit Delete	C8	Sumit Subba	9842155103	A2
Edit Delete	C9	Raju Shrestha	9842155214	A8
Edit Delete	C10	Dipesh Bhandari	9842155325	A3
Edit Delete	C11	Dhiraj Subedi	9842121498	A1

CUSTOMER_ID: C1
 CUSTOMER_NAME:
 Girwan Budhathoki
 CONTACT: 9842154326

Figure 104 Before editing value on Address id.



	CUSTOMER_ID	CUSTOMER_NAME	CONTACT	ADDRESS_ID
Update Cancel	C1	Girwan magad	9842154326	A11
Edit Delete	C4	Aakash dahal	9842154622	A3
Edit Delete	C5	Kabi Darlami	9842154770	A4
Edit Delete	C6	Ujwal Risal	9842154881	A9
Edit Delete	C7	Samyak Sakhya	9842154992	A1
Edit Delete	C8	Sumit Subba	9842155103	A2
Edit Delete	C9	Raju Shrestha	9842155214	A8
Edit Delete	C10	Dipesh Bhandari	9842155325	A3
Edit Delete	C11	Dhiraj Subedi	9842121498	A1

CUSTOMER_ID: C1
 CUSTOMER_NAME:
 Girwan Budhathoki
 CONTACT: 9842154326

Figure 105 While editing value on Address id.

Server Error in '/' Application.

ORA-02291: integrity constraint (NGAMESH.CUSTOMER_DELIVERY_ADDRESS_FK) violated - parent key not found

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.Data.OracleClient.OracleException: ORA-02291: integrity constraint (NGAMESH.CUSTOMER_DELIVERY_ADDRESS_FK) violated - parent key not found

Source Error:

An unhandled exception was generated during the execution of the current web request. Information regarding the origin and location of the exception can be identified using the exception stack trace below.

Stack Trace:

```
[OracleException (0x80131938): ORA-02291: integrity constraint (NGAMESH.CUSTOMER_DELIVERY_ADDRESS_FK) violated - parent key not found
]
System.Data.OracleClient.OracleConnection.CheckError(OciErrorHandle errorHandle, Int32 rc) +321
System.Data.OracleClient.OracleCommand.Execute(OciStatementHandle statementHandle, CommandBehavior behavior, Boolean needRowid, OciRowidDescriptor& rowidDescriptor, OciRowidDescriptor& rowidDescriptor) +524
System.Data.OracleClient.OracleCommand.ExecuteNonQueryInternal(Boolean needRowid, OciRowidDescriptor& rowidDescriptor) +524
System.Data.OracleClient.OracleCommand.ExecuteNonQuery() +102
System.Web.UI.WebControls.SqlDataSourceView.ExecuteDbCommand(DbCommand command, DataSourceOperation operation) +400
System.Web.UI.WebControls.SqlDataSourceView.ExecuteUpdate(IDictionary keys, IDictionary values, IDictionary oldValues) +704
System.Web.UI.DataSourceView.Update(IDictionary keys, IDictionary values, IDictionary oldValues, DataSourceViewOperationCallback callback) +106
System.Web.UI.WebControls.GridView.HandleUpdate(GridViewRow row, Int32 rowIndex, Boolean causesValidation) +1244
System.Web.UI.WebControls.GridView.HandleEvent(EventArgs e, Boolean causesValidation, String validationGroup) +917
System.Web.UI.WebControls.GridView.OnBubbleEvent(Object source, EventArgs e) +92
System.Web.UI.Control.RaiseBubbleEvent(Object source, EventArgs args) +37
System.Web.UI.WebControls.GridViewRow.OnBubbleEvent(Object source, EventArgs e) +97
```

Figure 106 After editing value on Address id.

Testing 20: Format violation.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Edit Delete	R3	Eats	541777
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R10	Boudha Restro	555110
Edit Delete	R9	Totala	544554

RESTAURANT_ID: R1
 AVAILABLE_ON_RESTAURANT:
 Restaurant Nepal
 TELEPHONE NO: 555111

Figure 107 Before editing value on telephone no.

	RESTAURANT_ID	AVAILABLE_ON_RESTAURANT	TELEPHONE_NO
Edit Delete	R1	Restaurant Nepal	555111
Edit Delete	R2	Kitchen Urlabari	555222
Update Cancel	R3	Eats	nineone
Edit Delete	R4	Test Of Nepal	555444
Edit Delete	R5	Momo Vibe	555555
Edit Delete	R6	Nepalese Flavor	555666
Edit Delete	R7	Grill Center	555777
Edit Delete	R8	Sekuwa Corner	555888
Edit Delete	R10	Boudha Restro	555110
Edit Delete	R9	Totala	544554

RESTAURANT_ID: R1
 AVAILABLE_ON_RESTAURANT:

Figure 108 While editing value on telephone no.

Server Error in '/' Application.

Input string was not in a correct format.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.FormatException: Input string was not in a correct format.

Source Error:

An unhandled exception was generated during the execution of the current web request. Information regarding the origin and location of the exception can be identified using the exception stack trace below.

Stack Trace:

```
[FormatException: Input string was not in a correct format.]
System.Number.StringToNumber(String str, NumberStyles options, NumberBuffer& number, NumberFormatInfo info, Boolean parseDecimal) +12838615
System.Number.ParseDecimal(String value, NumberStyles options, NumberFormatInfo numfmt) +133
System.Convert.ToDecimal(String value, IFormatProvider provider) +80
System.String.System.IConvertible.ToDecimal(IFormatProvider provider) +16
System.Convert.ChangeType(Object value, TypeCode typeCode, IFormatProvider provider) +505
System.Web.UI.WebControls.Parameter.GetValue(Object value, String defaultValue, TypeCode type, Boolean convertEmptyStringToNull, Boolean ignoreNullableTypeChanges) +68
System.Web.UI.WebControls.Parameter.GetValue(Object value, Boolean ignoreNullableTypeChanges) +68
System.Web.UI.WebControls.SqlDataSourceView.AddParameters(DbCommand command, ParameterCollection reference, IDictionary parameters, IDictionary values) +340
System.Web.UI.WebControls.SqlDataSourceView.ExecuteUpdate(IDictionary keys, IDictionary values, IDictionary oldValues, DataSourceViewOperationCallback callback) +1244
System.Web.UI.WebControls.GridView.HandleUpdate(GridViewRow row, Int32 rowIndex, Boolean causesValidation) +1244
System.Web.UI.WebControls.GridView.OnRowUpdating(GridViewRow row, Int32 rowIndex) +177
```

Figure 109 After editing value on telephone no.

15. Further Discussion

During the development of this I have imbibed the knowledge about the working of #C with its coding and designing. I got to explore .Net coding and designing furthermore. As I mentioned above .Net is people-oriented software, the future of coding will be undoubtedly involved vast designing. ASP.NET makes it much easier to manage web programming. ASP.NET keeps all the good features of classic ASP and improves on them. ASP.NET will undoubtedly be the tool of choice for most web developers.

During this coursework I have encountered many errors while integrating database in web application and I have overcome it by making different SQL command source on the aspx file. It was kind of hard and difficult to use the drag and drop features however it started giving me time on it and I found it every easy interface to make a web application. Editing the file after dragging needed container was a different level hardship as it was machine generated code and editing was very difficult at the beginning phase. As I started digging it more, I get quite familiar with its source code and the drag and drop features make it very easy to use.

References