

The Hang Seng University of Hong Kong  
MSIM 3201-Database Management(2021-2022)  
Team Project-Case Study

Topics: Option 1  
Platform Name: Food Niya Me  
Group: 1



S215359

Chan Hei Man

S215363

Chow Kwan Yee

S197671

Lau Hei Yee

S215366

Lee Ying Tung

# Content

1. Topic and objective
2. The revised version of Requirement specification
3. ERD design
  - a. Table
  - b. Schema
  - c. ERD graph
4. Display
  - a. SQL
  - b. NoSQL
5. Descriptions of Individual contributions

## Topic and Objective

We have created an All-in-one system(Food Niyame) for food delivery platforms to be our topic.

In our platform, we would display the information of different food delivery platforms, and the information on food, restaurants, restaurant take-out or to-go menu, payment methods and order information which have already been included in those food delivery platforms.

The new function that we added is to compare the price of different restaurants at the same time, allowing customers to compare the price of different restaurants from different platforms and order functions for customers to check order, also, as comment functions to reply.

## The revised version of the Requirement specification

For our system(Food Niyame), FoodPanda, OperRice and Deliveroo these three platforms have been chosen.

When customers open the system, they can choose the food by dish type, order platforms or the nearest restaurant of the customers. If customers chose the order platform, all the restaurants' information would be displayed.

After choosing the restaurant, the address and phone number would also be shown. When customers chose food, the system would give out the price from other platforms with the same restaurant, allowing customers to compare the total price order on different platforms.

Finally, the order information would be displayed before and after the customers pay for their reference.

## ERD design

### a. Table

FoodNiyame Table

Attribute Name	Data type
No.	INT
RestID	CHAR(8)
RestName	VARCHAR(500)
District	VARCHAR(100)
DishType	VARCHAR(200)
PriceRange	VARCHAR(100)
OrderPlatform	VARCHAR(255)
Foodpandadeliveryfee	DOUBLE
deliveroodeliveryfee	DOUBLE

RestaurantInfo Table

Attributes Name	Data type
RestID	CHAR(8)
RestName	VARCHAR(500)
RestaurantAddress	VARCHAR(500)
Restphone	VARCHAR(8)
Officialwebsite	TEXT

CustomerInfo

Attributes Name	Data type
CustID	CHAR(8)
CustFirstName	VARCHAR(100)
CustLastName	VARCHAR(150)
CustAddress	VARCHAR(500)
CustPhone	VARCHAR(8)
CustEmail	VARCHAR(255)

EmployeeInfo Table

Attributes Name	Data type
EmployeeID	CHAR(8)
EmployeeFirstName	VARCHAR(100)
EmployeeLastName	VARCHAR(150)
EmployeePhoneNo	INT
PlatformID	CHAR(8)
District	VARCHAR(100)

OrderInfo Table

Attributes Name	Data type
OrderID	CHAR(8)
OrdTime	TIME
OrdDate	DATE
CustID	CHAR(8)
RestID	CHAR(8)
FoodID	CHAR(8)
PlatformID	CHAR(8)
Special	VARCHAR(500)
OrignalPrice	DOUBLE
Discount	DOUBLE
TotalPrice	DOUBLE
paymentmethod	VARCHAR(50)
EmployID	CHAR(8)
Delivertime	TIME
DeliverDate	DATE
Remarks	TEXT
CustAdress	VARCHAR(500)

Comment Table

Attributes Name	Data type
CommNo	CHAR(8)
CommDate	DATE
CommTime	TIME
CustComn	TEXT
CustRate	INT
CustID	CHAR(8)
RestID	CHAR(8)
OrdID	CHAR(8)

DeliveryPlatform Table

Attributes Name	Data type
PlatformID	CHAR(8)
PlatName	VARCHAR(50)
Takeaway	VARCHAR(5)
Delivery	VARCHAR(5)

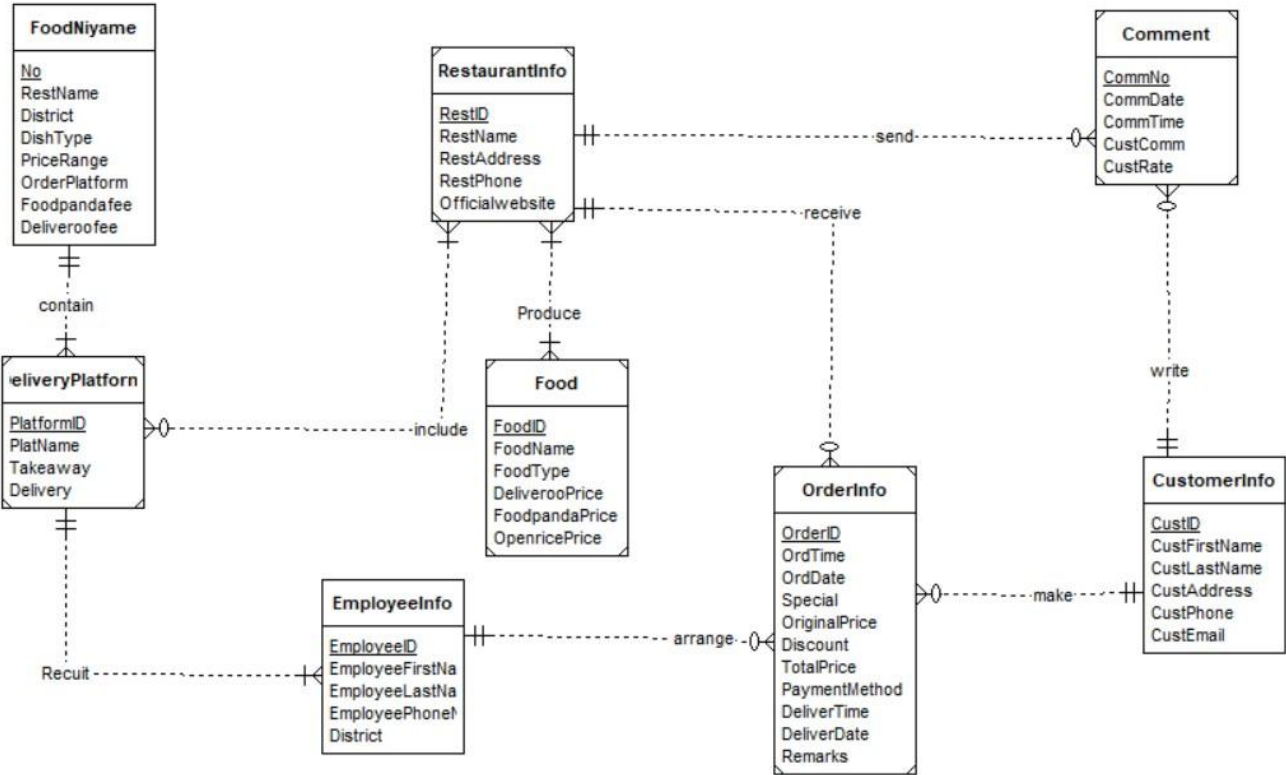
Food Table

Attributes Name	Data type
FoodID	CHAR(8)
FoodName	VARCHAR(500)
FoodType	VARCHAR(50)
Deliverooprice	DOUBLE
foodpandaprice	DOUBLE
Openriceprice	DOUBLE
RestName	VARCHAR(500)

## b. Schema

- FoodNiyema (No, RestID {FK}, RestName, District, DistType, PriceRange, Platform, Pandafee, DeliverooFee)
  - RestID refer to RestaurantInfo table
- RestaurantInfo (RestID, RestName, RestAddress, RestPhone, website)
- CustomerInfo (CustID, CustFirstName, CustLastName, CustAddress, CustPhone, CustEmail)
- EmployeeInfo (EmployeeID, EmployeeFirstName, EmployeeLastName, EmployeePhone, PlatformID {FK}, District)
  - PlatformID refer to DeliveryPlatform table
- Food (FoodID, FoodName, FoodType, DeliverooPrice, PandaPrice, OpenricePrice, RestName {FK})
  - RestName refer to the RestaurantInfo table
- OrderInfo (OrderID, OrdTime, OrdDate, CustID {FK}, RestID {FK}, FoodID {FK}, PlatformID {FK}, Special, OriginalPrice, Discount, Total, PaymentMethod, EmployeeID {FK}, DeliverTime, DeliverDate, Remarks)
  - CustID refer to CustomerInfo table
  - RestID refer to RestaurantInfo table
  - FoodID refer to Food table
  - PlatformID refer to DeliveryPlatform table
  - EmployeeID refer to EmployeeInfo table
- DeliveryPlatform ( PlatformID, PlatName, Takeaway, Delivery)
- Comment (CommNo, CommDate, CommTime, CustComm, CustRate, CustID {FK}, RestID {FK}, OrderID {FK})
  - CustID refer to CustomerInfo table
  - RestID refer to RestaurantInfo table
  - OrderID refer to OrderInfo table

c. ERD Graph





## Display

### a. SQL

(input data into database)

#### Example 1

Code:

```
INSERT orderinfo(OrderID, OrdTime, OrderDate, CustID, RestID, FoodID, PlatformID,
Special,
OriginalPrice, Discount, TotalPrice, paymentmethod, EmployID, Delivertime, DeliverDate,
Remarks)
VALUE ('O0000001', '17:00:00', '2022/04/24', 'C0000001', 'R0000002', 'F000002', 'D0000003',
'No', 180, 1, 180, 'visa', 'E000001', '17:35:01', '2022/04/24', 'I will go down stair to pick up.
Please contact me, thank you. '),
('O0000002', '13:00:05', '2022/04/25', 'C0000005', 'R0000029', 'F000145', 'D0000001',
'Seperate the source and chicken', 1018, 1, 1018, 'visa', 'E000056', '14:00:00', '2022/04/25',
'Please have contact me before deliver'),
('O0000003', '13:05:47', '2022/04/25', 'C0000002', 'R0000036', 'F000185', 'D0000003', 'Less
sugar', 397, 0.9, 357.3, 'master', 'E000020', '17:00:00', '2022/04/26', 'please deliver at 17:00 on
26/04/2022');
```

#### Example 2

```
INSERT foodniyame (No, RestID, RestName, District, DishType, PriceRange, OrderPlatform, Foodpandadeliveryfee
VALUE (53, 'R0000053', 'Hainanmom', 'Eastern', 'Chinese', '50-100', 'Openrice, Foodpanda', 20, null );
```

Code:

```
INSERT foodniyame (No, RestID, RestName, District, DishType, PriceRange, OrderPlatform,
Foodpandadeliveryfee, Deliveroodeliveryfee)
VALUE (53, 'R0000053', 'Hainanmom', 'Eastern', 'Chinese', '50-100', 'Openrice, Foodpanda', 20,
null );
```

(selecting order and output)

### Example 1

	CustID	OrderDate	FoodName	TotalPrice
1	C0000001	2022-04-24	Chicken Party for 2	180
2	C0000005	2022-04-25	Cold Vermicelli with Pork Chop	1018
3	C0000002	2022-04-25	Caramel Pudding Croissant	357.3

Code:

```
✓ Select x.CustID, x.OrderDate, y.FoodName, x.TotalPrice
  From orderinfo As x, Food As y
  Where x.FoodID=y.FoodID;
```

### Example 2

Code:

```
Select distinct x.RestID, x.RestName, x.PriceRange, y.RestaurantAddress,y.Restphone
From foodniyame As x, restaurantinfo As y
Where x.RestID=y.RestID AND RestaurantAddress Like '%Tai Po%';
```

	RestID	RestName	PriceRange	RestaurantAddress	Restphone
1	R0000003	McDonald	Below 50	Shop 9, 1/F, Uptown Plaza, No.9 Nam Wan Ro...	26529900
2	R0000005	Beans	50-150	Shop 670A-671B, G/F, Zone D, Tai Po Mega M...	53962747
3	R0000010	Tai Hing	50-80	Shop 13, G/F, Eightland Gardens, 2 On Chee...	26612711
4	R0000012	TamJai Yunnan Mixian	Below 50	Shop 027-030, 2/F, Zone A, Tai Po Mega Mal...	21690178
5	R0000017	Genki Sushi	Above 200	Shop A001, 1/F, Uptown Plaza, 9 Nam Wan Ro...	26766388
6	R0000026	Pho Le	51-100	Shop 518-520, 1/F, Zone C, Tai Po Mega Mal...	23368030
7	R0000030	Nene Chicken	101-200	Shop 287-295, 2/F, Zone B, Tai Po Mega Mal...	39710939

### Example 3

Code:

Select \*

From foodniyame

Where District = 'Eastern' and PriceRange Like 'Below 50';

	No	RestID	RestName	District	DishType	PriceRange	OrderPlatform
1	2	R0000002	KFC	Eastern	Western	Below 50	Foodpanda, Deliveroo
2	41	R0000041	Kung Fu Tea	Eastern	Drinks	Below 50	Deliveroo, Openrice

(updating data)

Example 1

Coding:

```
UPDATE food
  set FoodName= 'Wife Cakes'
  Where FoodID= 'F0000209';

Select *
  From food
  Where FoodID= 'F0000209';
```

Result:

FoodID	F0000209
FoodName	Wife Cakes
FoodType	cake
Deliverooprice	14
foodpandaprice	14
Openriceprice	9
RestName	Tai Cheong Bakery

b. noSQL

(insert data)

Example 1

```
> db.Food.insertMany([
...   {FoodID: "F0000271", FoodName: "HandSharded Chicken Rice (Spicy Green Papaya) with soup", FoodType: "Rice", foodpandaprice: "61", openriceprice: "45", RestName: "Hainanmom"},
...   {FoodID: "F0000272", FoodName: "Hainanese Boneless Chicken", FoodType: "dish", foodpandaprice: "77", openriceprice: "59", RestName: "Hainanmom"},
...   {FoodID: "F0000273", FoodName: "Beans Sprout Salad", FoodType: "snack", foodpandaprice: "42", openriceprice: "32", RestName: "Hainanmom"},
...   {FoodID: "F0000274", FoodName: "Sand Ginger Chicken with vegetable rice", FoodType: "rice", foodpandaprice: "46", openriceprice: "33", RestName: "Hainanmom"},
...   {FoodID: "F0000275", FoodName: "Hand Shredded Chicken Soba in Japanese Sasame Dressing/ Spicy Chilli Sauce/ Sweet & Sour Dressing", FoodType: "noodle", foodpandaprice: "48", deliverooprice: "37", RestName: "Hainanmom"}
... ]);
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("626cf0e6b8e2647b39883f69"),
    ObjectId("626cf0e6b8e2647b39883f6a"),
    ObjectId("626cf0e6b8e2647b39883f6b"),
    ObjectId("626cf0e6b8e2647b39883f6c"),
    ObjectId("626cf0e6b8e2647b39883f6d")
  ]
}
```

(finding data)

Example 1

```
> db.Food.find({RestName: "Hainanmom"}).pretty();
{
  "_id" : ObjectId("626cf0e6b8e2647b39883f69"),
  "FoodID" : "F0000271",
  "FoodName" : "HandSharded Chicken Rice (Spicy Green Papaya) with soup",
  "FoodType" : "Rice",
  "foodpandaprice" : "61",
  "openriceprice" : "45",
  "RestName" : "Hainanmom"
}
{
  "_id" : ObjectId("626cf0e6b8e2647b39883f6a"),
  "FoodID" : "F0000272",
  "FoodName" : "Hainanese Boneless Chicken",
  "FoodType" : "dish",
  "foodpandaprice" : "77",
  "openriceprice" : "59",
  "RestName" : "Hainanmom"
}
{
  "_id" : ObjectId("626cf0e6b8e2647b39883f6b"),
  "FoodID" : "F0000273",
  "FoodName" : "Beans Sprout Salad",
  "FoodType" : "snack",
  "foodpandaprice" : "42",
  "openriceprice" : "32",
  "RestName" : "Hainanmom"
}
{
  "_id" : ObjectId("626cf0e6b8e2647b39883f6c"),
  "FoodID" : "F0000274",
  "FoodName" : "Sand Ginger Chicken with vegetable rice",
  "FoodType" : "rice",
  "foodpandaprice" : "46",
  "openriceprice" : "33",
  "RestName" : "Hainanmom"
}
```

(updating data)

### Example 1

```
> db.Food.updateOne({_id:ObjectId("626ce99061adcf03bcc0ce55")}, {$set:{FoodName:"Wife Cakes"}});
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.Food.find({_id:ObjectId("626ce99061adcf03bcc0ce55")});
{ "_id" : ObjectId("626ce99061adcf03bcc0ce55"), "FoodID" : "F0000209", "FoodName" : "Wife Cakes", "FoodType" : "cake", "Deliverooprice" : 14, "foodpandaprice" : 14, "Openriceprice" : "9", "RestName" : "Tai Cheong Bakery" }
> db.Food.find({_id:ObjectId("626ce99061adcf03bcc0ce55")}).pretty();
{
  "_id" : ObjectId("626ce99061adcf03bcc0ce55"),
  "FoodID" : "F0000209",
  "FoodName" : "Wife Cakes",
  "FoodType" : "cake",
  "Deliverooprice" : 14,
  "foodpandaprice" : 14,
  "Openriceprice" : "9",
  "RestName" : "Tai Cheong Bakery"
}
```

## Descriptions of Individual contributions

Our system logo and PowerPoint picture sources are found on the website, they are decorated by Chan Hei Man.

For the design of the table and relationship, the convenience of finding the information is also considered during the design, these are done by all the group members. The research on food, price, restaurant address, restaurant website, platforms information, phone number, etc. is also done by all group members.

For the data input in databases and testing the display result such as foodinfo, customersinfo, restaurantinfo data, etc. and display of the result are done by Lau Hei Yee and Lee Ying Tung.

Our wonderful ERD diagrams are made by group members, together with finishing these diagrams, from the beginning to the end. We spend a lot of time fixing and updating the ERD until it is finished. The final version is done by Lau Hei Yee.

The report, PowerPoint and overall arrangement are done by Chan Hei Man and Chow Kwan Yee. The presenting video and demo were arranged by Lee Ying Tung.