ONLINE CATALOGUE SYSTEM FOR A FOOD PRODUCTION COMPANY

ABSTRACT

My project on catalogue system for a food production company is based on the data storage of the company. First of all a company will run with profit only when its data is secure and correct. My project is to save and update the data of the company. In a food production company the data of the availability of the food stock is very important for the customers. In company every customer or the user is given an identity for the clarity and for the upgradability of the company further production.

In my application I will maintain the total data of the company. I had made the different entities for the storage and upgrade of the data on food.

Coming to the payment of the stock or the items purchased is very secure and it is given an payment id for every payment that the user does and it is also stored securely.

Over all intension of this project is to save and update the data of the company.

REQUIREMENT ANALYSIS

List of tables:

USERS

FOODS

PAYMENTS

MANAGES

PAYS

List of attributes with their domain types:

USERS:

USER_EMAIL	VARCHAR2(25)
USER_ADDRESS	VARCHAR2(30)
USER_MOBILE	NUMBER(12)
USER_NAME	VARCHAR2(20)

FOODS:

FOOD_ID	VARCHAR2(20)
FOOD_NAME	VARCHAR2(30)
FOOD_USER_ID	VARCHAR2(20)
FOOD_TYPE	VARCHAR2(20)
FOOD_PRICE	NUMBER(5)

PAYMENTS:

PAY_DESC	VARCHAR2(30)
PAY_ID	VARCHAR2(20)
PAY_USER_ID	VARCHAR2(20)
PAY_DATE	DATE

PAYS:

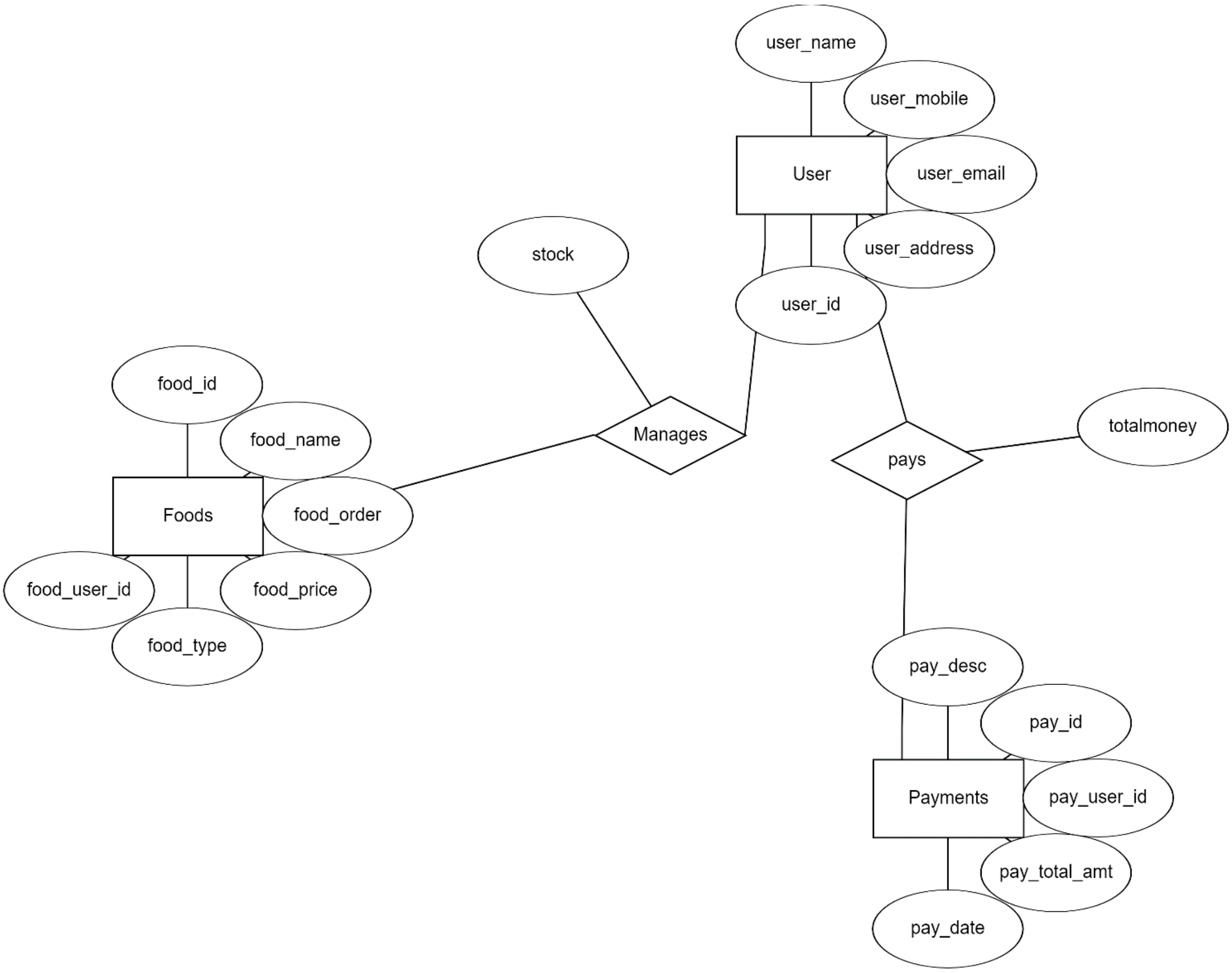
USER_ID	VARCHAR2(20)
PAY_ID	VARCHAR2(20)

TOTALAMT	NUMBER(10)

MANAGES:

USER_ID	VARCHAR2(20)
FOOD_ID	VARCHAR2(20)
STOCK	VARCHAR2(20)

ER DIAGRAM:



Mapping Cardinalities and Participation Constraints:

The relation "manages" between user and food is many to many because a user can manage different food products at the same time a food product can be managed by different users.

The relation "pays" between user and payments is one to many because one user can make different payments but a payment is made by only one user.

User has full participation in both the relations**.**

DDL COMMANDS:

```
SQL> create table foods(food_id varchar2(20),food_name varchar(30),food_user_id varchar2(20),food_type varchar(20),food_price number(5));


Table created.

SQL> create table payments(pay_desc varchar(30),pay_id varchar2(20),pay_user_id varchar2(20),pay_date date);


Table created.


SQL> create table users(user_id varchar2(20),user_email varchar(25),user_address varchar(30),user_mobile number(12),user_name varchar(20));


Table created.


SQL> create table manages(user_id varchar2(20),food_id varchar2(20),stock varchar(20));


Table created.


SQL> create table pays(user_id varchar2(20),pay_id varchar2(20),totalamt number(10));


Table created.


SQL> select *from tab;
```

```
TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
FOODS                          TABLE
MANAGES                        TABLE
PAYMENTS                       TABLE
PAYS                           TABLE
USERS                          TABLE

SQL> select *from foods;

no rows selected

SQL> desc foods;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 FOOD_ID                                            VARCHAR2(20)
 FOOD_NAME                                          VARCHAR2(30)
 FOOD_USER_ID                                       VARCHAR2(20)
 FOOD_TYPE                                          VARCHAR2(20)
 FOOD_PRICE                                         NUMBER(5)

SQL> alter table foods add primary key(food_id);

Table altered.

SQL> desc payments;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 PAY_DESC                                           VARCHAR2(30)
 PAY_ID                                    VARCHAR2(20)
 PAY_USER_ID                               VARCHAR2(20)
 PAY_DATE                                  DATE

SQL> alter table payments add primary key(pay_id);

Table altered.

SQL> desc manages;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USER_ID                                   VARCHAR2(20)
 FOOD_ID                                   VARCHAR2(20)
 STOCK                                     VARCHAR2(20)
SQL> alter table users add primary key(user_id);

Table altered.

SQL> alter table manages add foreign key(user_id) references users;

Table altered.

SQL> alter table manages add foreign key(food_id) references foods;

Table altered.

SQL> alter table manages add primary key(user_id,food_id);

Table altered.
```

```
SQL> alter table pays add foreign key(pay_id) references payments;


Table altered.
```

DML COMMANDS:

```
SQL> insert into foods
values('&food_id','&food_name','&food_user_id','&food_type',&food_price);
Enter value for food_id: 1
Enter value for food_name: pasta
Enter value for food_user_id: 18
Enter value for food_type: junk
Enter value for food_price: 500
old   1: insert into foods
values('&food_id','&food_name','&food_user_id','&food_type',&food_price)
new   1: insert into foods values('1','pasta','18','junk',500)

1 row created.

SQL> /
Enter value for food_id: 2
Enter value for food_name: pizza
Enter value for food_user_id: 19
Enter value for food_type: main
Enter value for food_price: 250
old   1: insert into foods
values('&food_id','&food_name','&food_user_id','&food_type',&food_price)
new   1: insert into foods values('2','pizza','19','main',250)

1 row created.

SQL> /
Enter value for food_id: 3
Enter value for food_name: coffee
Enter value for food_user_id: 20
Enter value for food_type: hot drink
Enter value for food_price: 150
old   1: insert into foods
values('&food_id','&food_name','&food_user_id','&food_type',&foo
d_price)
new   1: insert into foods values('3','coffee','20','hot drink',150)

1 row created.

SQL> /
Enter value for food_id: 4
Enter value for food_name: icecream
Enter value for food_user_id: 21
Enter value for food_type: desert
Enter value for food_price: 200
old   1: insert into foods
values('&food_id','&food_name','&food_user_id','&food_type',&foo
d_price)
new   1: insert into foods values('4','icecream','21','desert',200)

1 row created.

SQL> desc users;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USER_ID                                   NOT NULL VARCHAR2(20)
 USER_EMAIL                                         VARCHAR2(25)
 USER_ADDRESS                                       VARCHAR2(30)
 USER_MOBILE                                        NUMBER(12)
 USER_NAME                                          VARCHAR2(20)
```

```
SQL> insert into users('&user_id','&user_email','&user_address',&user_mobile,'&user_name');
SQL> insert into users values('&user_id','&user_email','&user_address',&user_mobile,'&user_name');
Enter value for user_id: 18-01
Enter value for user_email: kaka@gmail.com
Enter value for user_address: bombay
Enter value for user_mobile: 9876543210
Enter value for user_name: kaka
old   1: insert into users values('&user_id','&user_email','&user_address',&user_mobile,'&user_name')
new   1: insert into users values('18-01','kaka@gmail.com','bombay',9876543210,'kaka')

1 row created.

SQL> /
Enter value for user_id: 18-02
Enter value for user_email: kiku@gmail.com
Enter value for user_address: hyderabad
Enter value for user_mobile: 8907654321
Enter value for user_name: kiku
old   1: insert into users values('&user_id','&user_email','&user_address',&user_mobile,'&user_name')
new   1: insert into users values('18-02','kiku@gmail.com','hyderabad',8907654321,'kiku')

1 row created.

SQL> /
```

```
Enter value for user_id: 18-03
Enter value for user_email: kaki@gmail.com
Enter value for user_address: banglore
Enter value for user_mobile: 6789012345
Enter value for user_name: kaki
old   1: insert into users
values('&user_id','&user_email','&user_address',&user_mobile,'&us
er_name')
new   1: insert into users values('18-
03','kaki@gmail.com','banglore',6789012345,'kaki')

1 row created.

SQL> /
Enter value for user_id: 18-04
Enter value for user_email: kiki@gmail.com
Enter value for user_address: delhi
Enter value for user_mobile: 7890123456
Enter value for user_name: kiki
old   1: insert into users
values('&user_id','&user_email','&user_address',&user_mobile,'&us
er_name')
new   1: insert into users values('18-
04','kiki@gmail.com','delhi',7890123456,'kiki')

1 row created.

SQL> desc payments;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 PAY_DESC                                           VARCHAR2(30)
 PAY_ID                                    NOT NULL VARCHAR2(20)
 PAY_USER_ID                                        VARCHAR2(20)
 PAY_DATE                                           DATE
```

```
SQL> insert into payments
values('&pay_desc','&pay_id','&pay_user_id','&pay_date');
Enter value for pay_desc: thankyou
Enter value for pay_id: 10-01
Enter value for pay_user_id: 1
Enter value for pay_date: 01-jan-2020
old   1: insert into payments
values('&pay_desc','&pay_id','&pay_user_id','&pay_date')
new   1: insert into payments values('thankyou','10-01','1','01-jan-2020')

1 row created.

SQL> /
Enter value for pay_desc: welcome
Enter value for pay_id: 10-02
Enter value for pay_user_id: 2
Enter value for pay_date: 19-oct-2019
old   1: insert into payments
values('&pay_desc','&pay_id','&pay_user_id','&pay_date')
new   1: insert into payments values('welcome','10-02','2','19-oct-2019')

1 row created.

SQL> /
Enter value for pay_desc: visitonceagain
Enter value for pay_id: 10-03
Enter value for pay_user_id: 3
Enter value for pay_date: 23-feb-2019
old   1: insert into payments
values('&pay_desc','&pay_id','&pay_user_id','&pay_date')
```

```
new   1: insert into payments values('visitonceagain','10-03','3','23-feb-2019')

1 row created.

SQL> /
Enter value for pay_desc: vanakam
Enter value for pay_id: 10-04
Enter value for pay_user_id: 4
Enter value for pay_date: 15-aug-2018
old   1: insert into payments values('&pay_desc','&pay_id','&pay_user_id','&pay_date')
new   1: insert into payments values('vanakam','10-04','4','15-aug-2018')

1 row created.

SQL> desc manages
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USER_ID                                   NOT NULL VARCHAR2(20)
 FOOD_ID                                   NOT NULL VARCHAR2(20)
 STOCK                                              VARCHAR2(20)

SQL> insert into manages values('&user_id','&food_id','&stock');
Enter value for user_id: 18-01
Enter value for food_id: 1
Enter value for stock: 1 unit left
old   1: insert into manages values('&user_id','&food_id','&stock')
new   1: insert into manages values('18-01','1','1 unit left')

1 row created.

SQL> /
```

```
Enter value for user_id: 18-02
Enter value for food_id: 2
Enter value for stock: out of stock
old   1: insert into manages values('&user_id','&food_id','&stock')
new   1: insert into manages values('18-02','2','out of stock')

1 row created.

SQL> /
Enter value for user_id: 18-03
Enter value for food_id: 3
Enter value for stock: 3 units left
old   1: insert into manages values('&user_id','&food_id','&stock')
new   1: insert into manages values('18-03','3','3 units left')

1 row created.

SQL> /
Enter value for user_id: 18-04
Enter value for food_id: 4
Enter value for stock: full stock
old   1: insert into manages values('&user_id','&food_id','&stock')
new   1: insert into manages values('18-04','4','full stock')

1 row created.

SQL> desc pays;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USER_ID                                   NOT NULL VARCHAR2(20)
 PAY_ID                                    NOT NULL VARCHAR2(20)
 TOTALAMT                                           NUMBER(10)

SQL> insert into pays values('&user_id','&pay_id',&totalamt);
```

```
Enter value for user_id: 18-01
Enter value for pay_id: 10-01
Enter value for totalamt: 10000
old   1: insert into pays values('&user_id','&pay_id',&totalamt)
new   1: insert into pays values('18-01','10-01',10000)

1 row created.

SQL> /
Enter value for user_id: 18-02
Enter value for pay_id: 10-02
Enter value for totalamt: 20000
old   1: insert into pays values('&user_id','&pay_id',&totalamt)
new   1: insert into pays values('18-02','10-02',20000)

1 row created.

SQL> /
Enter value for user_id: 18-03
Enter value for pay_id: 10-03
Enter value for totalamt: 45000
old   1: insert into pays values('&user_id','&pay_id',&totalamt)
new   1: insert into pays values('18-03','10-03',45000)

1 row created.

SQL> /
Enter value for user_id: 18-04
Enter value for pay_id: 10-04
Enter value for totalamt: 60000
old   1: insert into pays values('&user_id','&pay_id',&totalamt)
new   1: insert into pays values('18-04','10-04',60000)

1 row created.
```

```
SQL> desc pays;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USER_ID                                   NOT NULL VARCHAR2(20)
 PAY_ID                                    NOT NULL VARCHAR2(20)
 TOTALAMT                                           NUMBER(10)

SQL> select *from pays;

USER_ID              PAY_ID                 TOTALAMT
-------------------- -------------------- ----------
18-01                10-01                     10000
18-02                10-02                     20000
18-03                10-03                     45000
18-04                10-04                     60000
```