# 1 系统需求分析

## 系统具体需求

分析对象:大型酒店和大型餐饮服务

分析流程：客户点餐—>厨房服务—>上菜等全流程的分析

存在的问题：上菜紊乱，没有秩序。出现的问题包括如下：

1. 没有完备的餐饮管理流程。
2. 会出现有的厨师非常空闲没有菜去做，而有些厨师则非常的忙，降低了上菜的效率
3. 会出现有的服务员非常的空闲，没有去上菜，而有些服务员则连续跑腿，一人上好几桌子的菜，造成了上菜的效率非常的低下。
4. 一旦顾客食用出现问题，不知道问题出在哪个环节，缺少相应的厨师的做菜记录，以及服务员的服务过程的记录。出现问题则找不出问题所在，而影响到整个餐饮企业的发展。

**为解决上述的种种问题，以及提高用户的用餐体验以及提高大型酒店以及大型餐馆的服务本小组为将要设计开发的数据库，设计出了如下的数据流程图对各个环节进行监控的把控。**

**该点餐系统需要实现的功能如下：**

1. 用户的注册
2. 用户的点餐
3. 生成订单信息
4. 厨师的接单，并生成生成做菜单号，修改厨师的状态
5. 服务员接单，生成服务员接单信息，修改服务员的状态
6. 查询某个用户所食用的某种菜的菜号是由哪个厨师做的，以及哪个服务员上的。

## 1.2 系统业务流程分析

### 1.3.1 点餐过程以及任务分配

（1）用户注册ID

（2）用户通过所在的餐桌号，以及自己的ID号，以及菜单的ID号进行点餐，点餐

（3）点餐后生成一份订单信息表，根据点餐信息表，厨师进行接单，并将接单信息存放到厨师的接单信息表中，以及厨师的状态

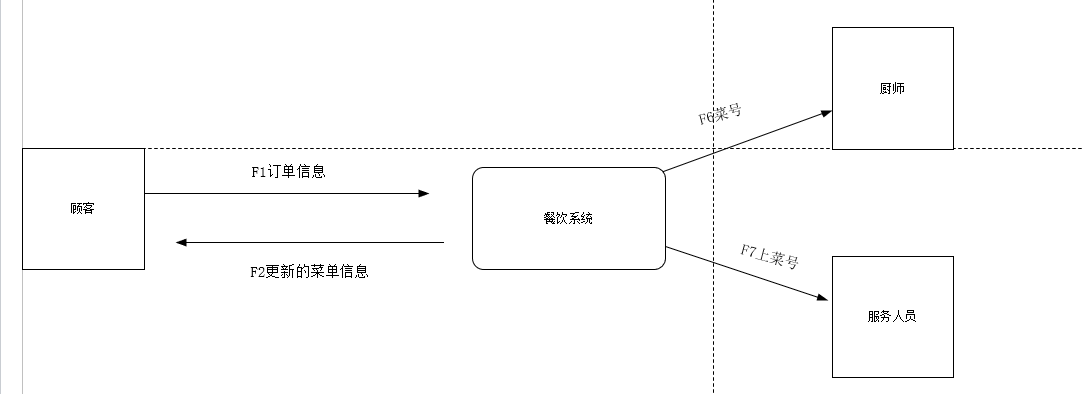
（4）服务员进行接单，并将接单的生成一条接单信息，存到服务员接单信息表中，并修改服务员的状态。

（5）厨师和服务员都接了改订单后修改订单的状态

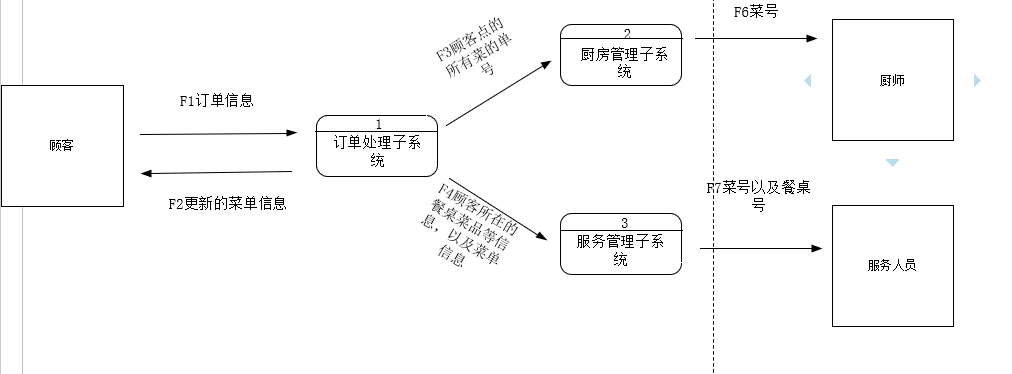
系统数据流图

### 1.3.1 顶层数据流图

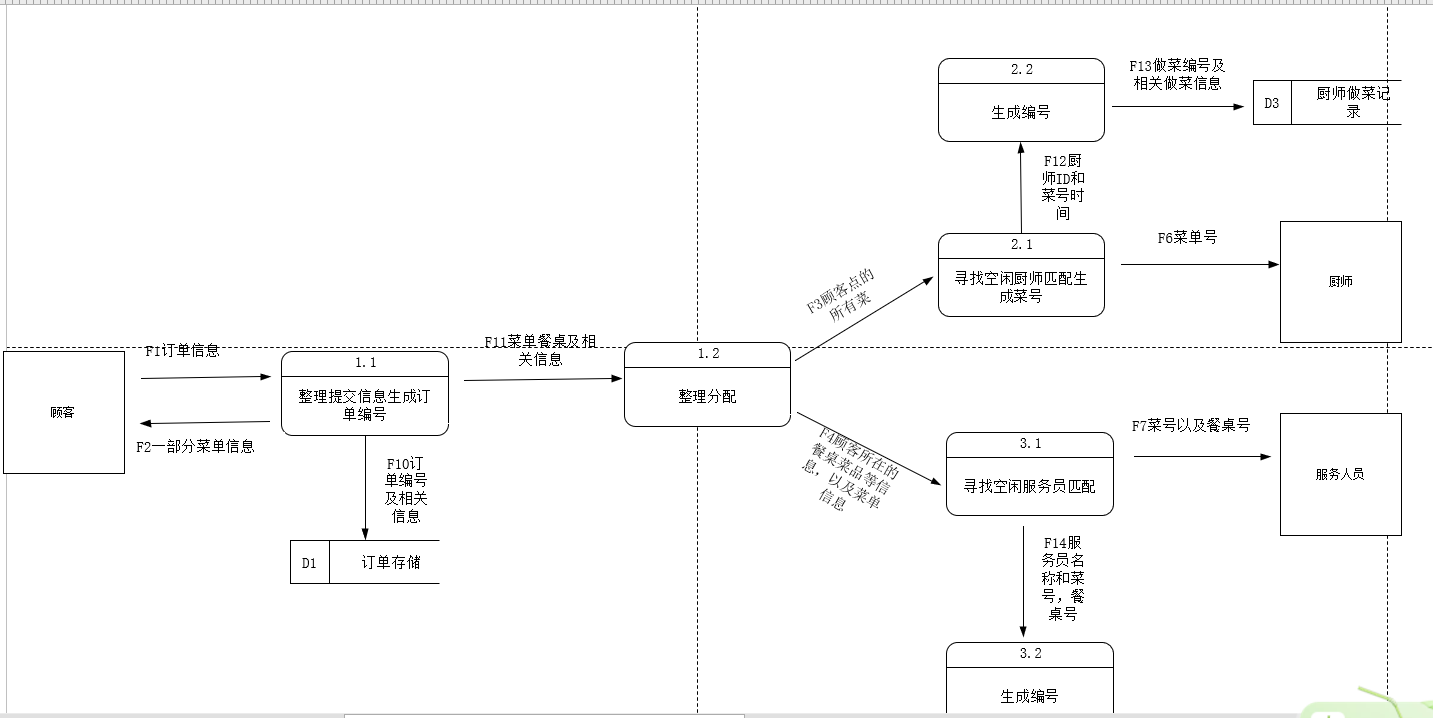
**顶层：**



**二层：**

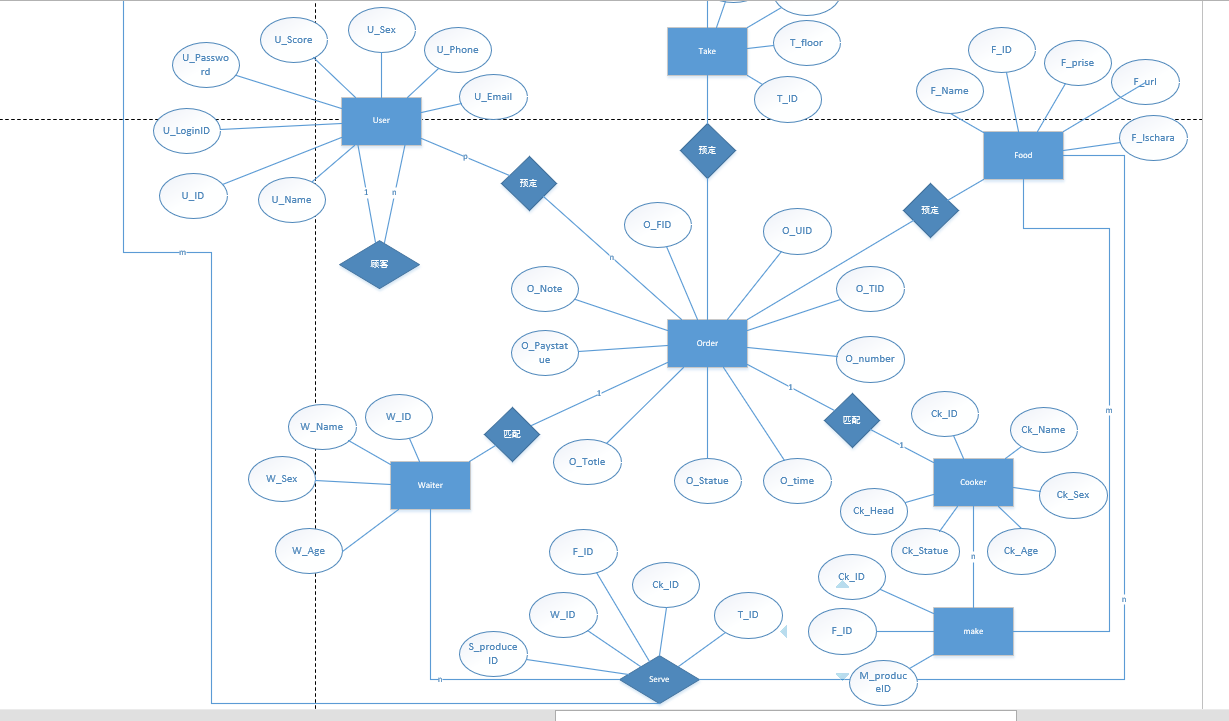


### 1.3.2 零层数据流图 ：



# 2 数据库设计

## 2.1 概念结构设计（ER图）



在上述图中，部分对应关系解释如下：

(1)

一个用户ID可以被多个顾客使用，但是一个顾客只能注册一个用户ID 1：n

（2）

一个用户可以下达多份订单，多份订单也可以由一个用户下达p:n

（3）一个餐桌可以对应多份订单，一份订单也可以对应多个餐桌 m:n

（4）一份菜可以被多张订单预订，多张订单也可以预订一份菜

（5）一个餐桌可以的菜可以由多个服务员来上多个菜，一个服务员也可以上多个餐桌的多个菜，一份菜可以上多个餐桌，也可以由多个服务员上。n:p:m

(6) 一个厨师可以做多份菜，一份菜也由多个厨师做。n:m

## 2.2 逻辑结构设计

### 2.2.1 表清单

|  |  |
| --- | --- |
| **实体名称** | **中文注释** |
| User | 用户表 |
| Food | 菜单表 |
| Take | 餐桌表 |
| Order | 订单表 |
| Coker | 厨师表 |
| Make | 厨师做菜表 |
| Waiter | 服务员表 |
| Server | 服务员上菜表 |

### 2.2.2 表设计

（1）Task表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **数据类型** | **长度** | **允许空** | **主键** | **说明** |
| T\_ID | int | 11 | no | No | 任务编号 |
| T\_time | Datetime |  | No | no | 任务的下达时间 |
| T\_statue | varchar | 20 |  | no | 任务的状态 |
| T\_Component\_id | int |  | No | Yes | 物资编号 |
| T\_component\_name | varchar | 20 | No | No | 物资的描述 |
| T\_jiliang | varchar | 10 |  |  | 计量 |
| T\_number | int |  |  |  | 采购数量 |
| T\_find\_time | Datetime |  | No | No | 物资采购时间 |

Worker表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **数据类型** | **长度** | **允许空** | **主键** | **说明** |
| W\_ID | int | 6 | no | yes | 员工的编号 |
| W\_name | Varchar | 10 | yes | no | 员工的姓名 |
| W\_post | varchar | 10 | yes | no | 员工的职位 |
| W\_sex | Bit | 1 | yes | no | 性别，0代表女生，1代表男生 |
| W\_boss | varchar | 10 | no | no | 员工的领导 |
| W\_iphone | varchar | 11 | no | no | 员工的手机号码 |

Inquiry表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **数据类型** | **长度** | **允许空** | **主键** | **说明** |
| T\_component\_id | int | 6 | no | Yes(同时是任务表中的外键) | 物资的编号 |
| I\_name | Varchar | 100 | yes | no | 物资描述 |
| I\_jiliang | varchar | 10 | no | no | 物资的计量单位 |
| I\_max | float |  | yes | no | 物资的最高价 |
| I\_min | Float |  | yes | no | 物资的最低价 |
| I\_number | int |  | no | no | 采购的数量 |
| I\_price | Float |  | no | no | 最近一次报价 |

Support表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **数据类型** | **长度** | **允许空** | **主键** | **说明** |
| S\_ID | int | 6 | no | yes | 供应商的编号 |
| S\_name | Varchar | 10 | No | no | 供应商的姓名 |

Price\_table（审价）表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **数据类型** | **长度** | **允许空** | **主键** | **说明** |
| **T\_component\_id** | **int** |  | **no** | **Yes(做任务单的外键)** | **物资的编号** |
| S\_ID | int |  | no | Yes（商家表的外键） | 商家的编号 |
| S\_name | varchar | 20 | no | no | 商家的名称 |
| W\_ID | int |  | No | Yes（员工表的外键） | 询价员的编号 |
| Pt\_time | Datetime |  | No | No | 审价时间 |
| Pt\_number | int |  | No | no | 数量 |
| Pt\_price | int |  | No | no | 物资的单价 |
| Pt\_total | float |  | no | No | 金额 |

Agreement表

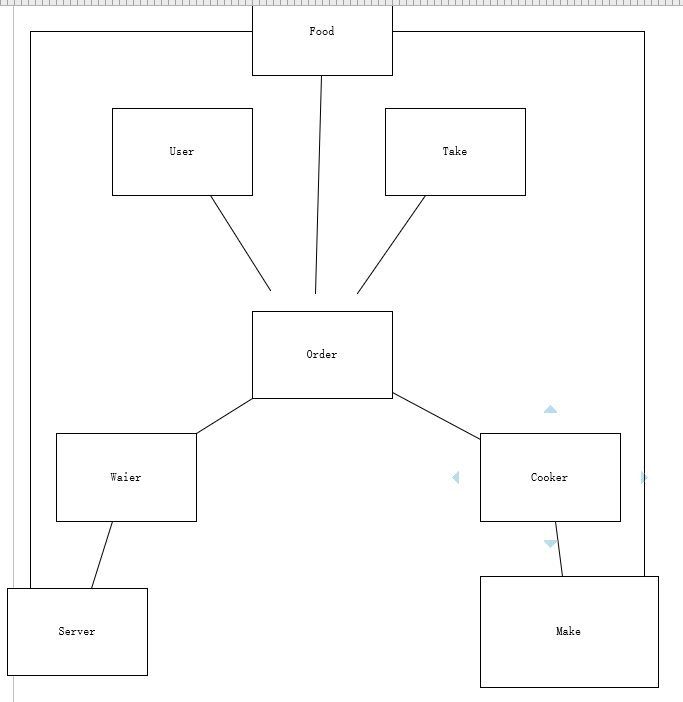
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **数据类型** | **长度** | **允许空** | **主键** | **说明** |
| A\_ID | int |  | no | yes | 合同的编号 |
| S\_ID | int |  | no | YES(商家的外键) | 供应商 编号 |
| W\_ID | Int | 20 | No | YES(员工的编号当外键) | 员工编号 |
| T\_component\_id | int | 20 | no | Yes(物资编号当外键) | 采购人员 |
| A\_jiliang | varchar | 10 | no |  | 计量 |
| A\_price | Float |  | No |  | 单价 |
| A\_number | int |  |  |  | 采购的数量 |
| A\_total | float |  | no | no | 总价格 |
| A\_project\_id | int |  | No |  | 工程编号 |
| A\_project\_name | varchar | 20 | No |  | 工程名称 |

Bill表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **数据类型** | **长度** | **允许空** | **主键** | **说明** |
| **T\_component\_id** | **int** |  | **no** | **Yes（当做外键）** | **物资的编号** |
| **S\_ID** | **int** |  | **no** | **Yes（当做外键）** | **商家的编号** |
| B\_ID | int |  | no | Yes（当做外键） | 发票的编号 |
| B\_time | Datetime |  | no | no | 发票的时间 |
| B\_type | varchar | 20 | no | No | 发票的类型 |
| B\_tax\_price | float |  | no | No | 税额 |
| B\_tax | float |  | No | no | 价税合计 |
| B\_Doing | Varchar | 20 | no | No | 经办人 |

# 3 数据库应用系统开发

## 3.1 表创建（表关系图）



## 3.2 索引创建（可以截图）

| **索引名** | **代码** |
| --- | --- |
| PK\_\_U\_ser\_\_5A2040DB571DF1D5 |  |
| PK\_\_T\_ake\_\_83BB1FB260A75C0F |  |
| PK\_\_Food\_\_2C6EC7C3656C112C |  |
| PK\_\_O\_rder\_\_FBDAD26117036CC0 |  |
| PK\_\_Cooker\_\_C36D020C6B24EA82 |  |
| PK\_\_Waiter\_\_8175B5737F2BE32F |  |
| PK\_\_Make\_\_30CF4A3B797309D9 |  |
| PK\_\_S\_erver\_\_D0B3BD9104E4BC85 |  |

## 3.3 视图创建

| **视图名（作用）** | **代码** |
| --- | --- |
| 无 |  |

## 3.4 触发器创建

| **触发器名（作用）** | **代码** |
| --- | --- |
| 无 |  |

# 4 功能模块设计

| **功能名称** | **模块名称** |
| --- | --- |
| 用户注册和点餐 | 用户注册(ZC)，并将注册信息存到U\_ser  表中 |
| 用户点餐（DC），并将点餐信息存到O\_rder表中 |
| 厨师接单 | 厨师接单（CSJD），并将接单信息存到Make表中 |
| 服务员接单 | 服务员接单（FWYJD）,并将接单信息存到S\_erver表中 |

# 5功能模块实现

## 5.1 功能语句

### 5.1.1 用户注册

|  |  |
| --- | --- |
| 语句 | create proc ZC  @U\_ID int ,  @U\_name varchar(20),  @U\_loginID varchar(30),  @U\_password varchar(20),  @U\_score int ,  @U\_sex varchar(5),  @U\_Email varchar(50),  @U\_iphone varchar(11)  AS  begin  insert into U\_ser(U\_ID,U\_name,U\_loginID,U\_password,U\_score,U\_sex,U\_Email,U\_iphone)  values(@U\_ID,@U\_name,@U\_loginID,@U\_password,@U\_score,@U\_sex,@U\_Email,@U\_iphone)  end  Go  Exec ZC 9226,'林永健','1499755237','123456',100,'男','1499755237@qq.com','18012822313' |
| 执行结果 | 执行上诉存储过程，生成一条用户信息在U\_ser表：    查询U\_ser表： |

### 5.1.2 点餐过程

|  |  |
| --- | --- |
| 语句 | create proc DC  @F\_ID varchar(100),  @U\_ID int,  @T\_ID varchar(20),  @O\_ID varchar(50),  @O\_number int,  @O\_note varchar(50) ----备注  as  declare  @O\_total float, --总价格  @O\_paystatue Bit, ---1代表已经支付，代表未支付  @O\_statue varchar(10), ---订单的状态  @F\_Price float,  @O\_time date  begin  select @F\_Price=F\_price from Food where F\_ID=@F\_ID  set @O\_statue='未接单'  set @O\_paystatue=1  set @O\_total= @F\_Price\*@O\_number  set @O\_time=GETDATE()  insert into O\_rder(F\_ID,U\_ID,T\_ID,O\_ID,O\_time,O\_statue,O\_number,O\_total,O\_paystatue,O\_note)  values(@F\_ID,@U\_ID,@T\_ID,@O\_ID,@O\_time,@O\_statue,@O\_number,@O\_total,@O\_paystatue,@O\_note)  end  Go  exec DC '1号',9226,'WZ005','LB#001',10,'不吃辣' |
| 执行结果 | 执行：    查询： |

### 5.1.3 厨师接单

|  |  |
| --- | --- |
| 语句 | create proc CSJD  @CK\_ID varchar(20),  @F\_ID varchar(100),  @M\_ProduceID varchar(30)  as  begin  if not exists(select \*from O\_rder where F\_ID=@F\_ID)  begin  print '客户并没有点这个菜'  rollback  end  insert into Make(CK\_ID,F\_ID,M\_ProduceID)  values(@CK\_ID,@F\_ID,@M\_ProduceID)  update Cooker set CK\_statue='忙' where CK\_ID=@CK\_ID  end  GO  exec CSJD 'CS001','1号','M001' |
| 执行结果 | 执行：        查询： |

### 5.1.4 服务员接单

|  |  |
| --- | --- |
| 语句 | create proc FWYJD  @W\_ID varchar(20),  @T\_ID varchar(20),  @F\_ID varchar(100),  @S\_ProduceID varchar(30)  as  begin  if not exists(select \*from O\_rder where T\_ID=@T\_ID)  begin  print'你选择上菜的餐桌有误';  rollback;  end  if not exists(select \*from O\_rder where F\_ID=@F\_ID)  begin  print '顾客并没有点这个菜';  rollback;  end  insert into S\_erver(W\_ID,T\_ID,F\_ID,S\_ProduceID)  values(@W\_ID,@T\_ID,@F\_ID,@S\_ProduceID)  update O\_rder set O\_statue='已接单'  update Waiter set W\_statue='忙' where W\_ID=@W\_ID  end  GO  exec FWYJD 'FW001','WZ005','1号','SC001' |
| 执行结果 | 查询： |