

一：引言

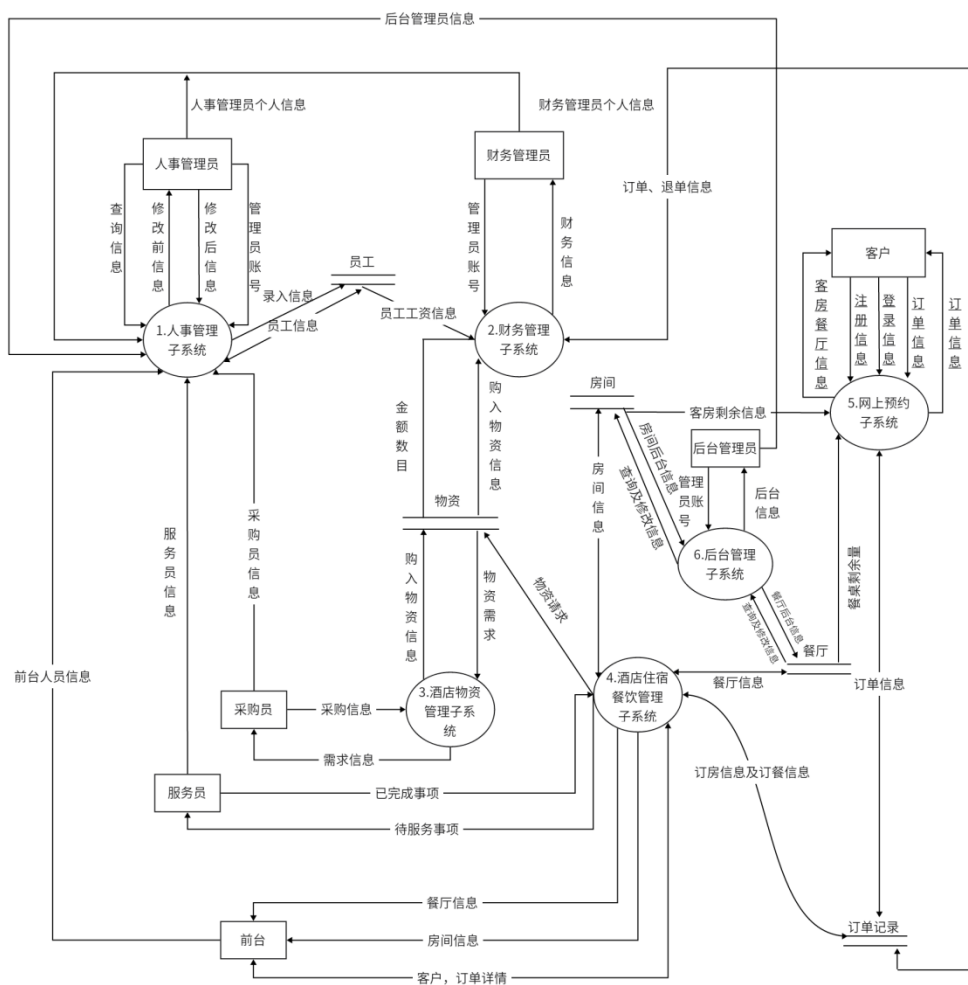
编写本文档主要用于说明关于酒店管理系统的具体系统架构设计。其中，具体说明了系统架构设计的目标和约束需求，规定了系统的开发环境和运行环境，阐述了逻辑架构设计的整体思想和具体设计的实现，介绍了系统的主要业务流程和系统的功能模块的划分以及模块之间的联系，表明了系统采用存储的数据库类型以及数据库又采用了什么设计编辑工具和数据库中存储的各种表的具体信息，最后是设计了系统核心功能的界面原型。

本文档是对系统架构设计的详细说明，通过编写该系统架构设计说明书，有利于了甲方更好地理解系统逻辑架构和系统相关功能，也便于系统管理员更好地管理系统以及在出现问题后能够更好地对系统进行维护。

二：体系结构设计

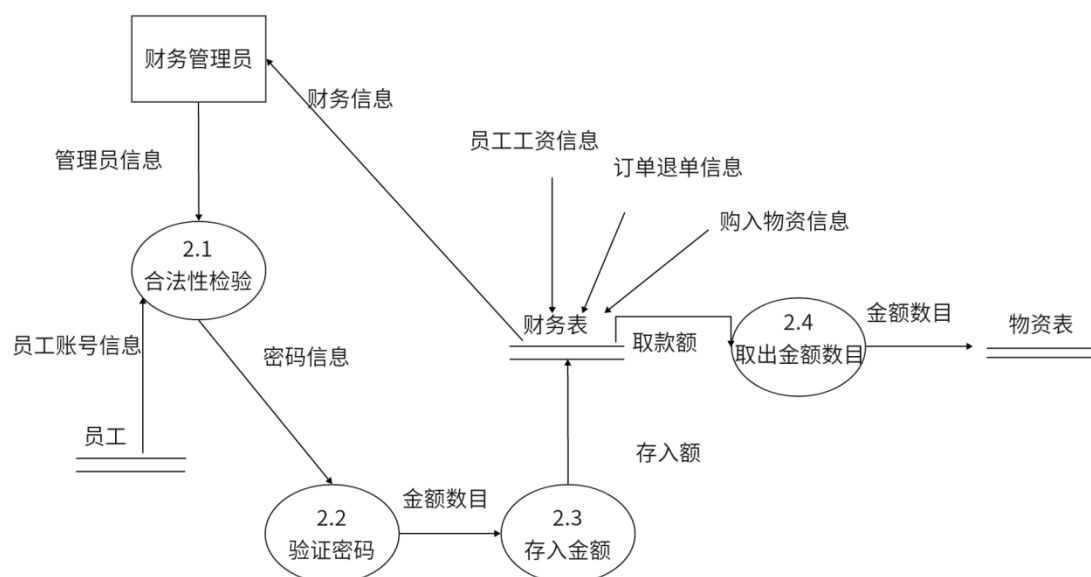
2.1 改善后的数据流图

2.1.1 第一层数据流图

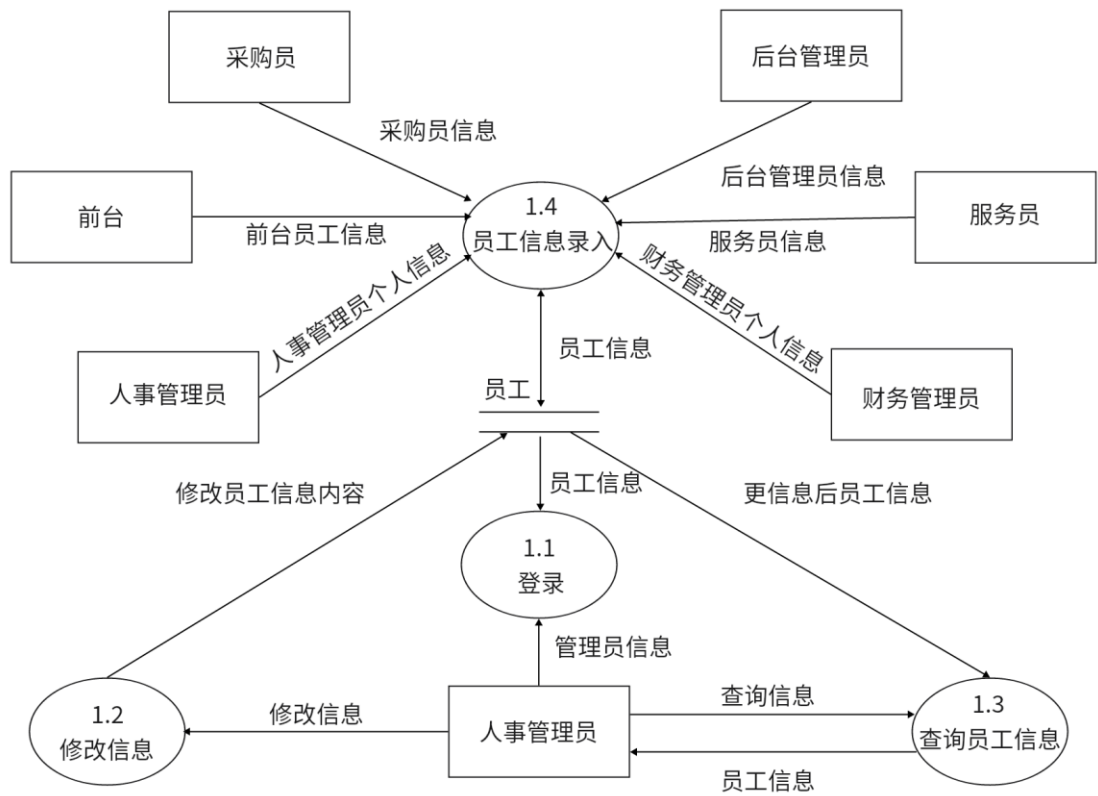


2.1.2 第二层数据流图

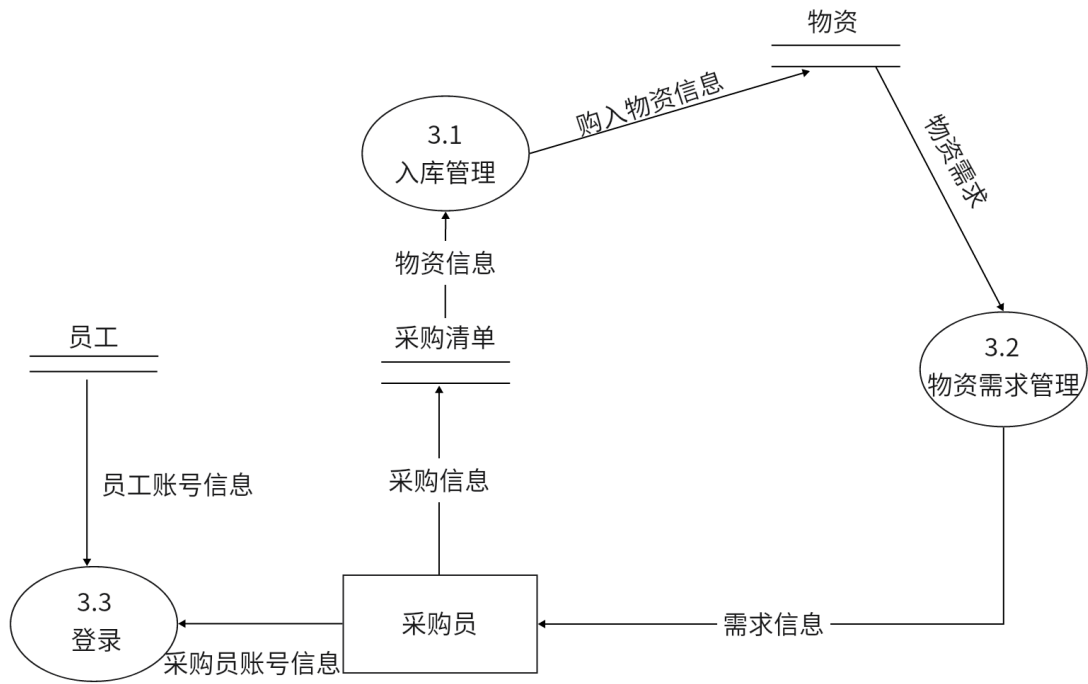
① 财务管理子系统



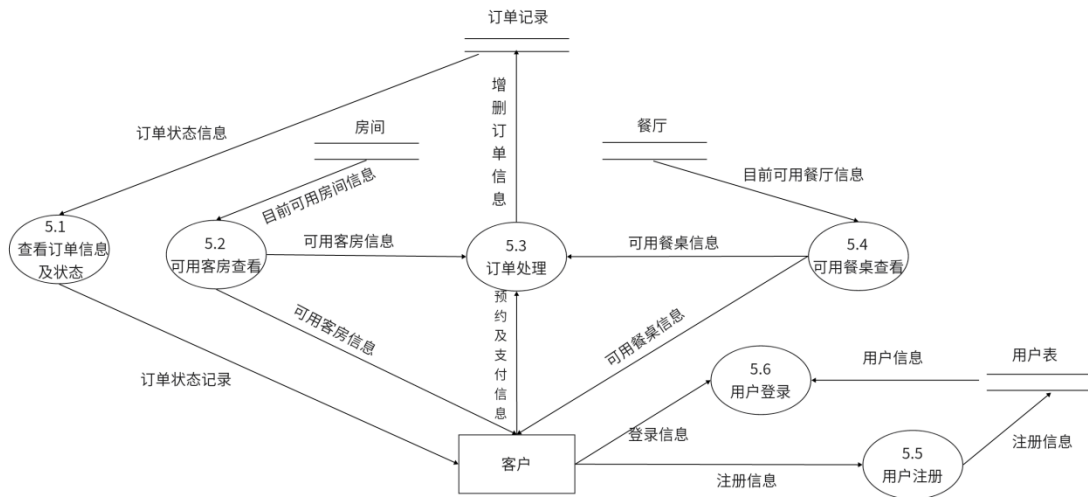
② 人事管理子系统



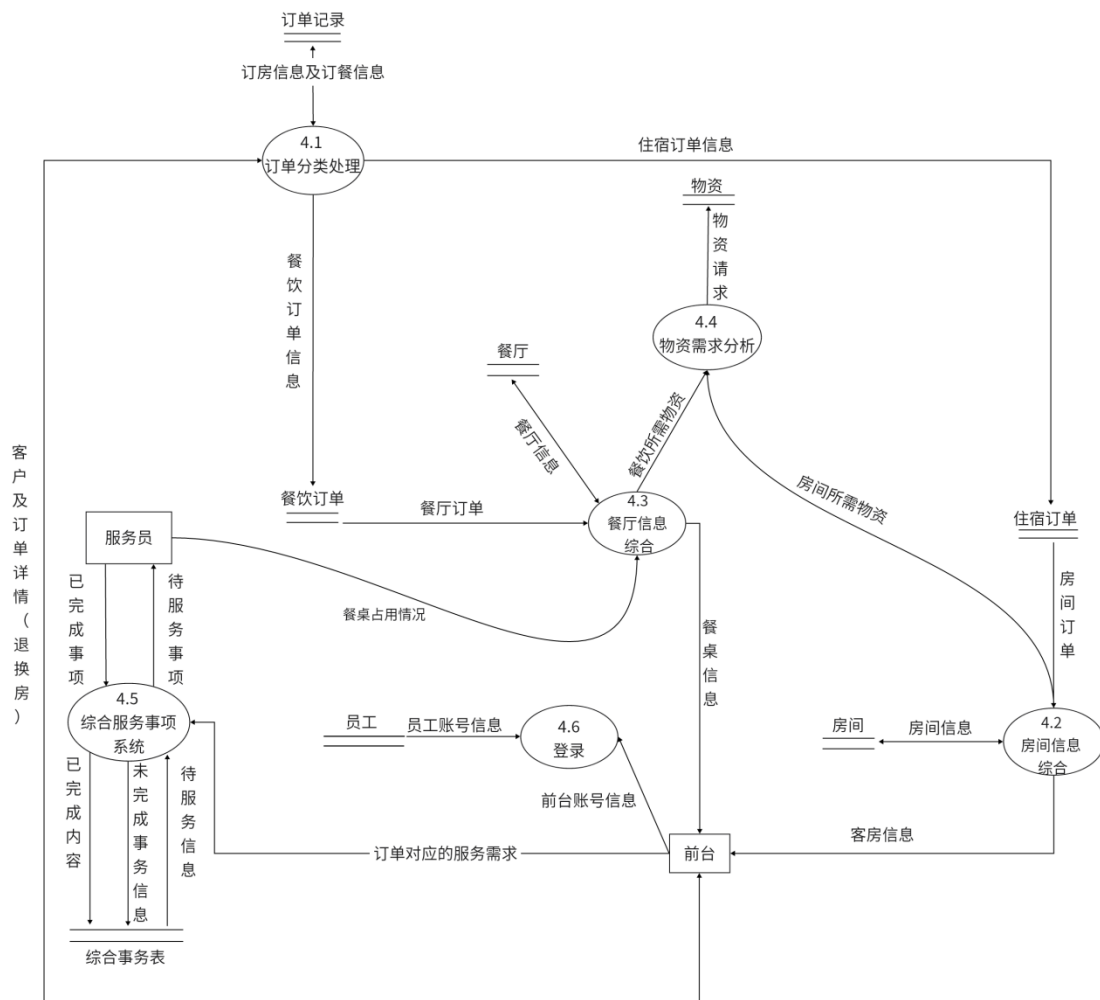
③ 物资管理子系统



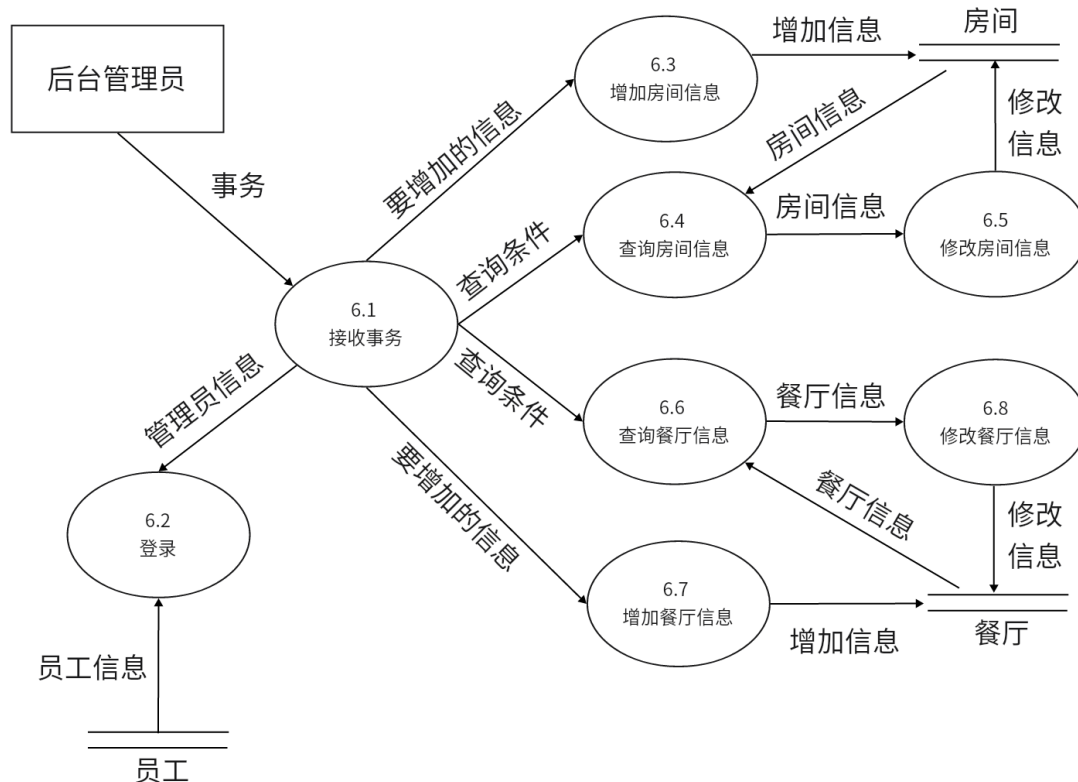
④ 网上预约子系统



⑤ 酒店住宿管理子系统



⑥ 后台管理子系统



2.2 系统功能模块划分

(1) 网上预约子系统

① 用户登陆注册

用户可以通过网页进行账号的注册和登陆，打开网站想要进行操作时，如果没有账号则要注册一个，若之前有账号则可以直接登陆

② 可用客房查询

用户可以输入需要预定的客房，查询该客房情况，也可以查询总共的客房信息，决定自己是否预约客房。

③ 可用餐桌查询

用户入住办理成功后，用户可以输入需要预定的餐桌，查询该餐桌情况，也可以查询总共的餐桌信息，决定自己是否预约餐桌。

④ 查询订单状态

用户可以通过系统查询自己所下的订单状态，从而决定是否取消订单，增加

订单，修改订单。

⑤ 预约房间/取消预约

用户可以通过系统查看房间信息，预约自己心仪的房间，也可以在规定时间内，取消对房间的预约，系统会根据订单情况判断是否可以取消预约。

(2) 物资管理子系统

① 采购员采购物资

采购员可以通过系统采购采购清单上的物资

② 采购员查看物资需求

采购员可以通过系统查看酒店物资的需求情况，根据需求来下订单

③ 采购员打采购清单

采购员可以将需要采购的物品打印出采购清单，系统会自动将清单上的物资入库

(3) 人事管理子系统

① 员工信息录入

人事管理员可以通过系统录入员工信息

② 查询员工信息

人事管理员可以通过系统查询员工信息

③ 修改员工信息

人事管理员可以通过系统修改员工信息

(4) 后台管理子系统

① 增删查改房间信息

后台管理员可以通过系统增删改查房间信息

②增删查改餐厅信息

后台管理员可以通过系统增删改查餐厅信息

③接收事务

后台管理员可以通过系统接收需要做的事务，然后再进行实际操作

(5) 财务管理子系统

①查询财务明细

财务管理员可以通过系统查询财务账单明细，明确金额的收支。

②修改财务表（采购信息导入/工资发放）

财务管理员可以将支出明细添加到财务单中，或者向财务表中添加收入金额账单

(6) 酒店住宿餐厅管理子系统

①前台直接办理入住

顾客可以通过前台办理入住，前台查询相应的房间，为用户选择合适的房间，并将入住信息录入，并将住宿订单存入订单表中，然后让用户缴纳相应的房间费用和押金，并将订单传入财务表中

②前台办理已预约用户入住

已预约顾客可以通过前台，将订单编号给前台查看，前台确认信息后将顾客信息录入，前台办理入住手续并将订单信息进行修改。

③用户退房

用户正常退房，返还用户订金，并将退还的订金订单写入财务表，若用户提前退房，则退还一定房费，若用户延迟退房，则根据延迟时间决定是否罚钱或者默认续住，用户离开后，记录用户真正离开的时间，并将退还定金信息写入

数据库

④用户换房

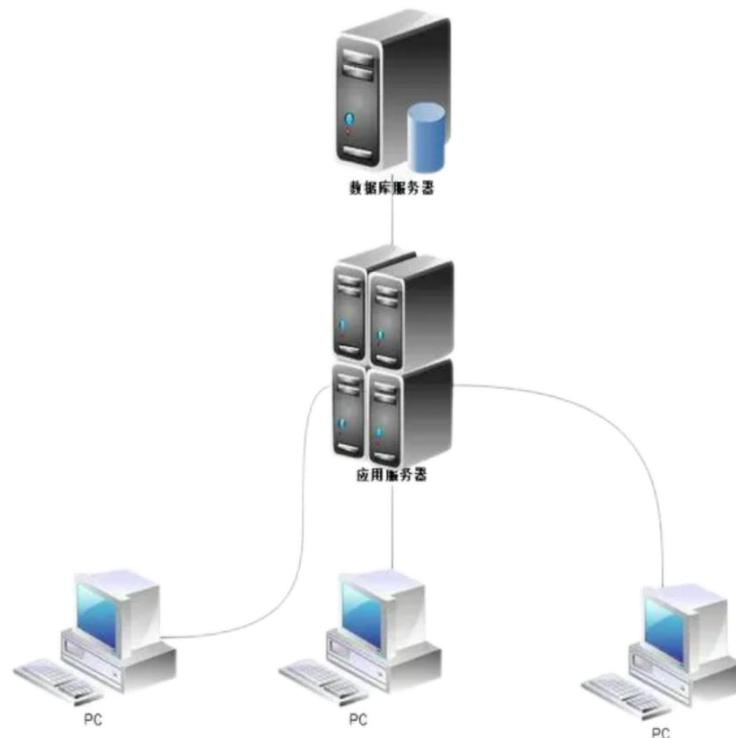
用户可以根据自己需求更换套房，如果能够更换，前台将之前的住宿订单删除，在添加新的房间订单即可，金额只需要补充或退还之间差值

⑤用户续住

用户可以通过前台进行续住，前台需查询该房间在续住时间段是否有人预定，之后将信息反馈给用户，若房间空闲，即可为用户办理续住。

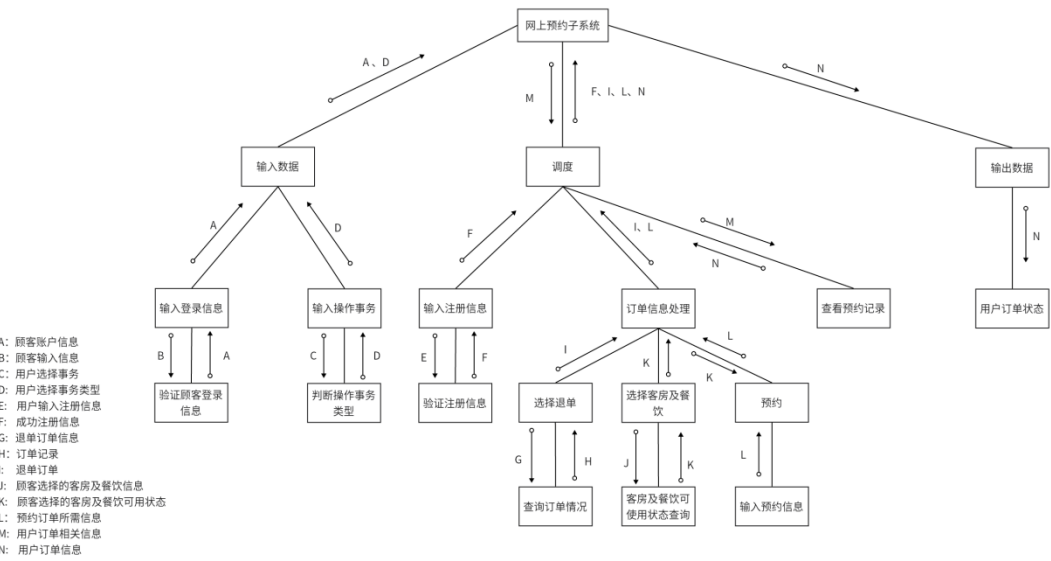
2.3 系统架构图

为使开发流程顺利进行，本次项目采用 Vue 框架开发的前端页面，采用 IntelliJ IDEA 作为开发工具，Tomcat 和 JSP 作为后台服务器，MySQL 作为数据库。

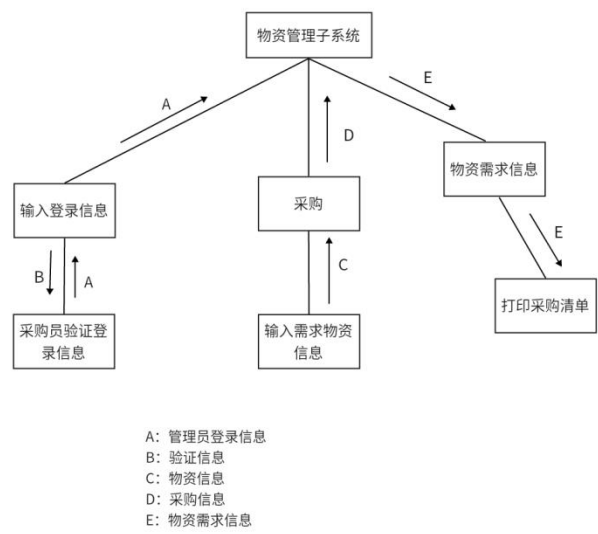


2.4 系统结构图

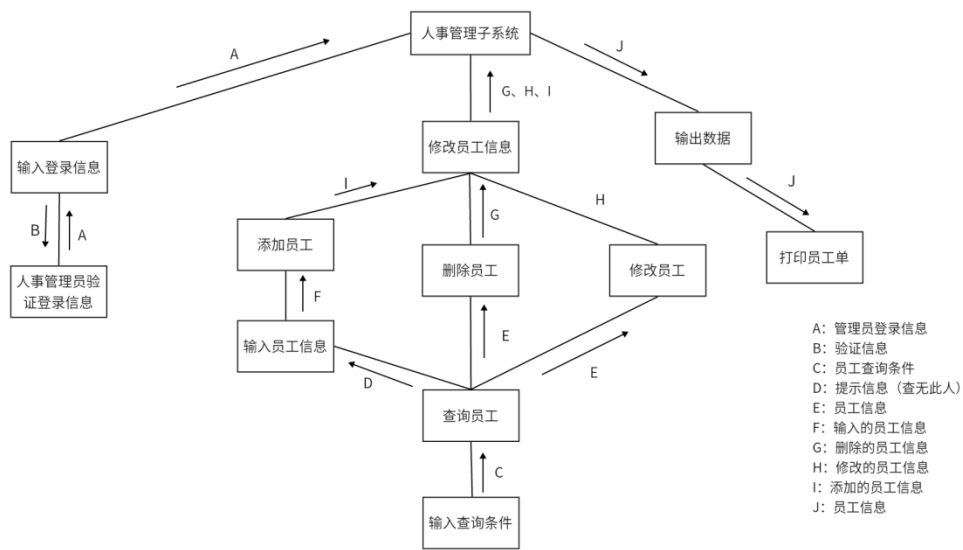
(1) 网上预约子系统系统结构图



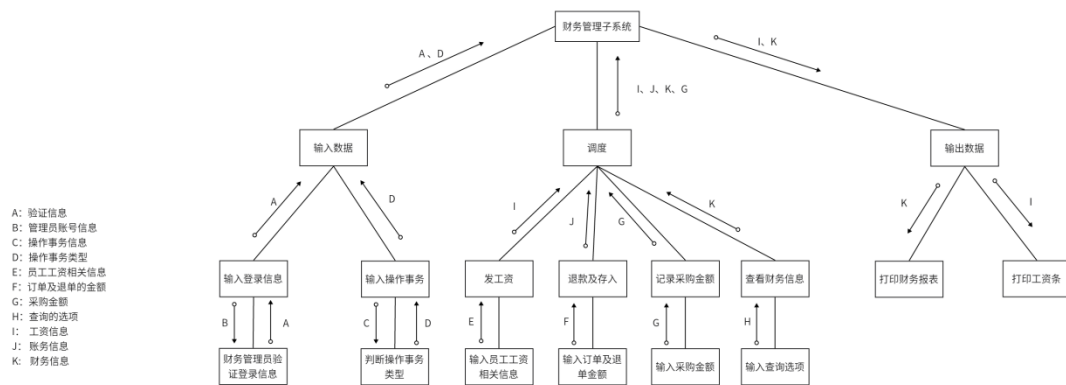
(2) 物资管理子系统系统结构图



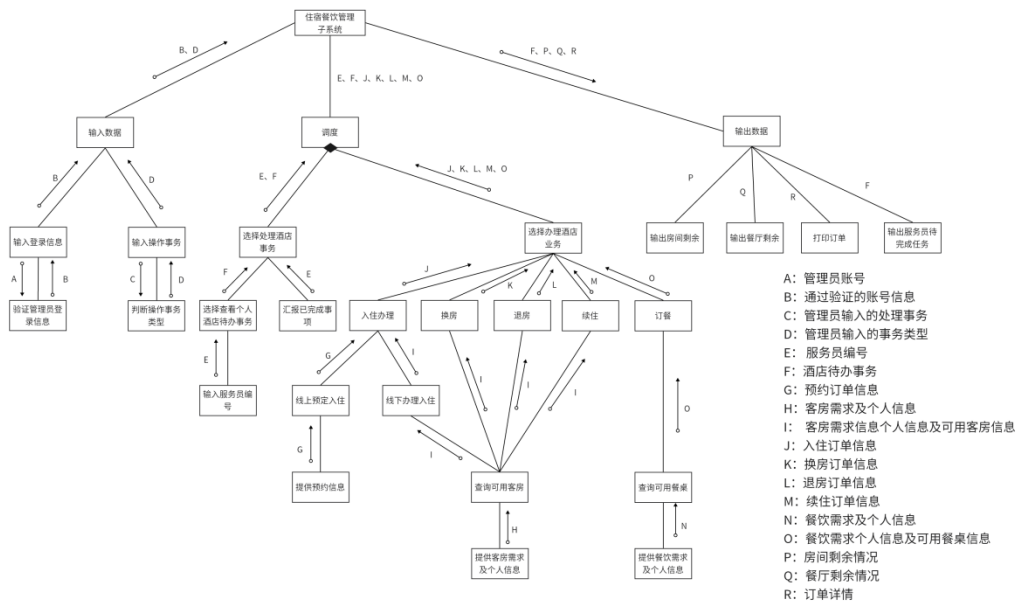
(3) 人事管理子系统系统结构图



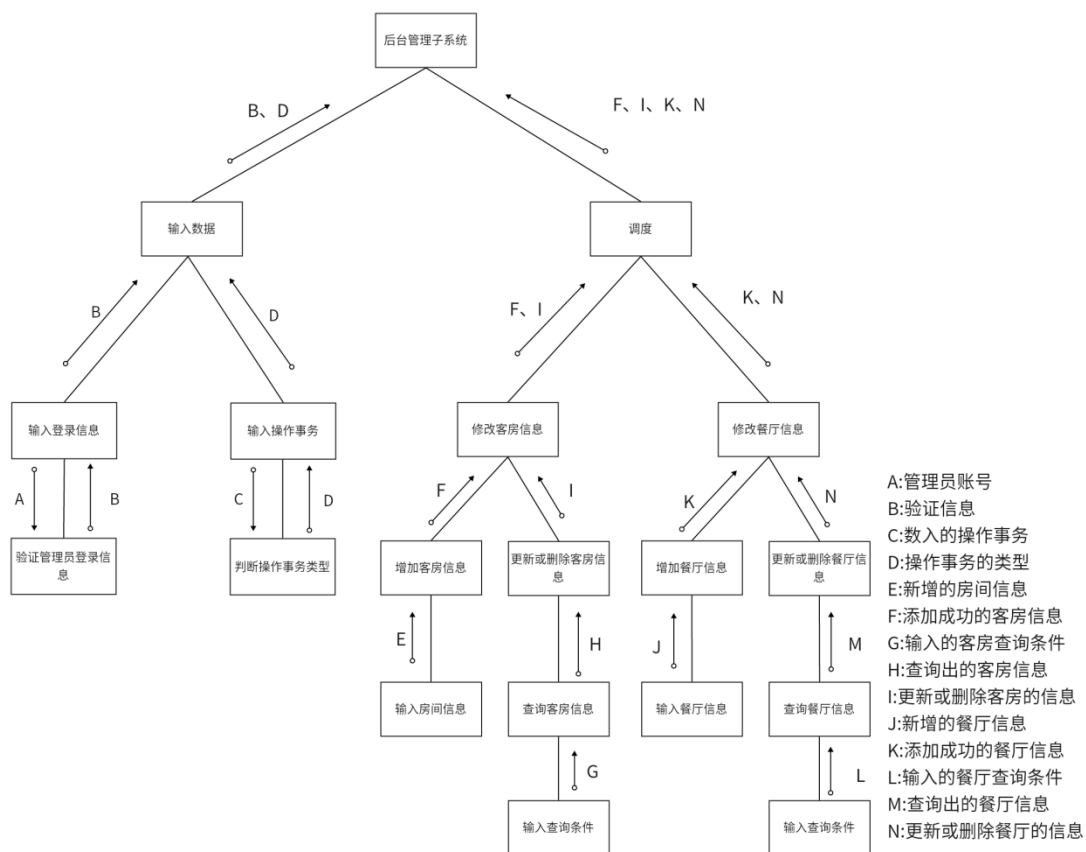
(4) 财务管理子系统系统结构图



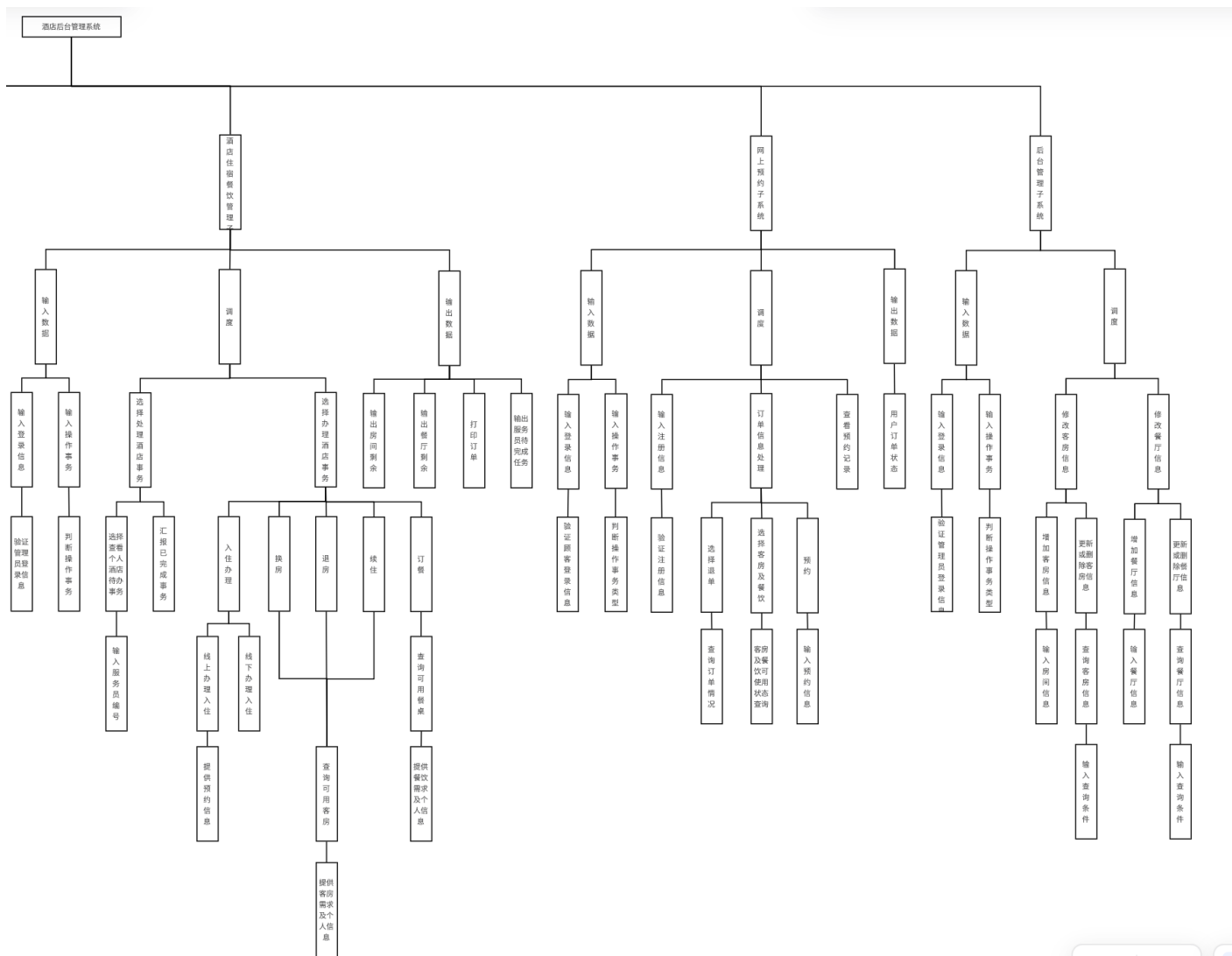
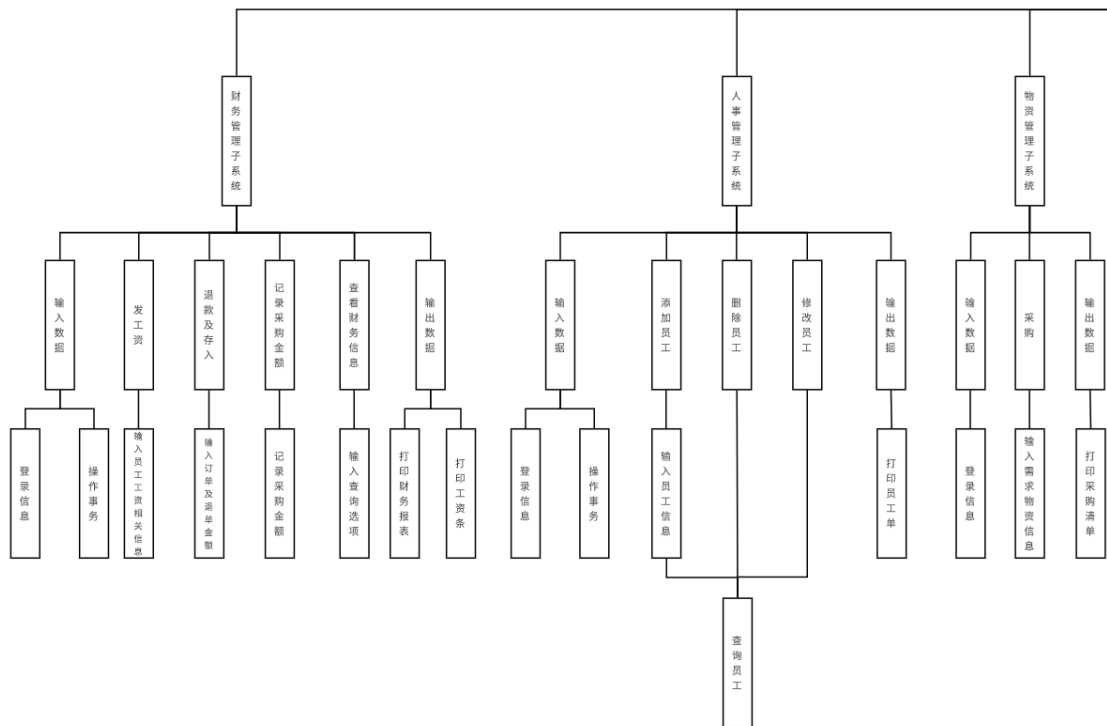
(5) 住宿管理子系统



(6) 后台管理子系统



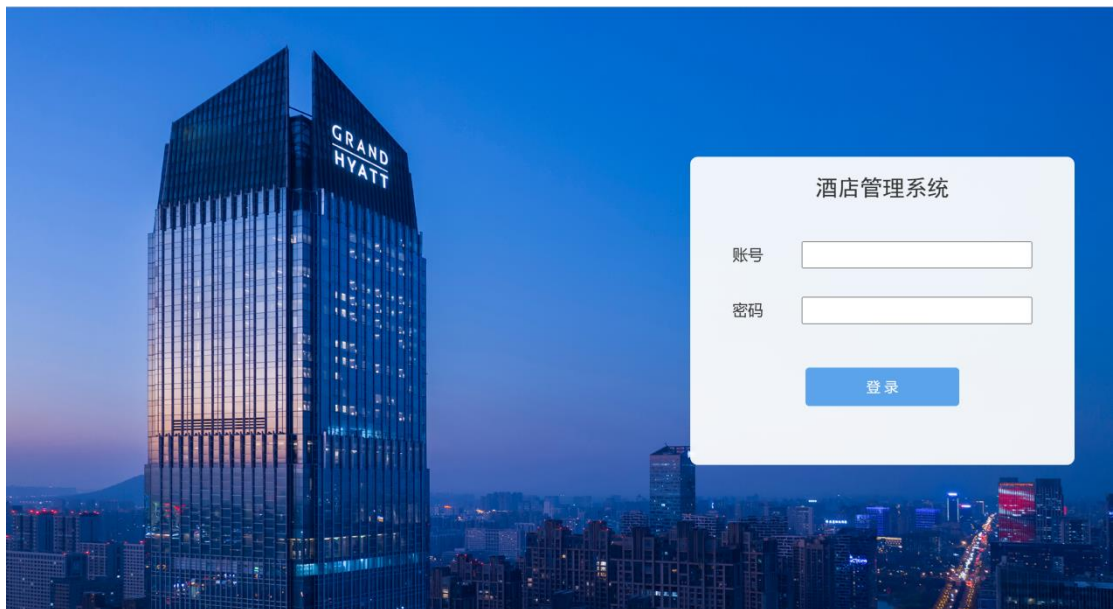
(7) 总体系结构图



三：接口设计

3.1 人机交互接口设计

①管理员登录界面



②可用房间信息查询界面



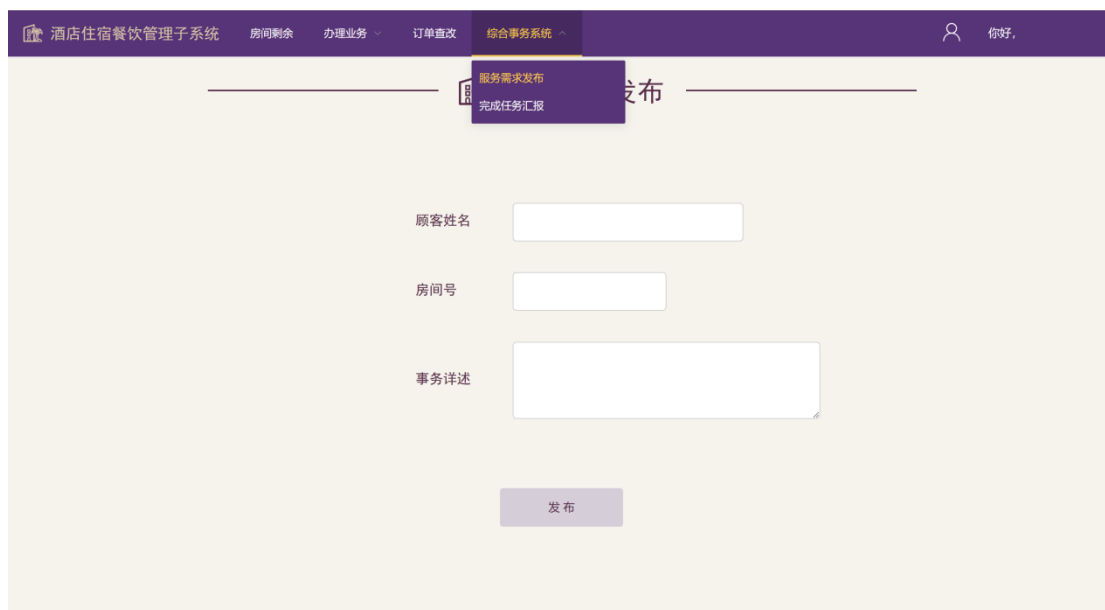
③前台办理线下入住界面



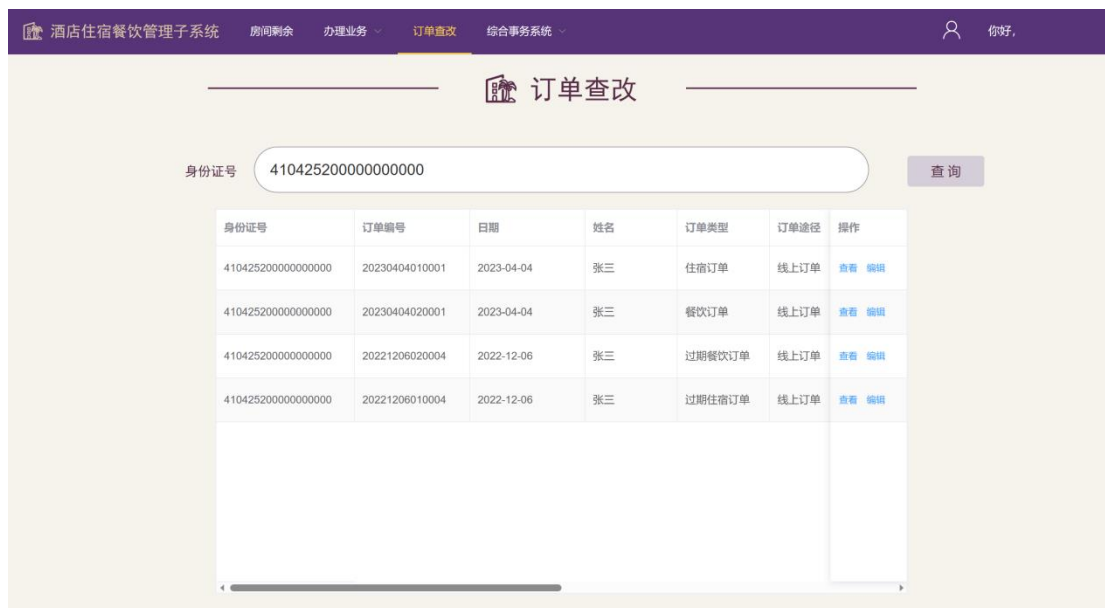
④前台查询餐厅余位信息界面



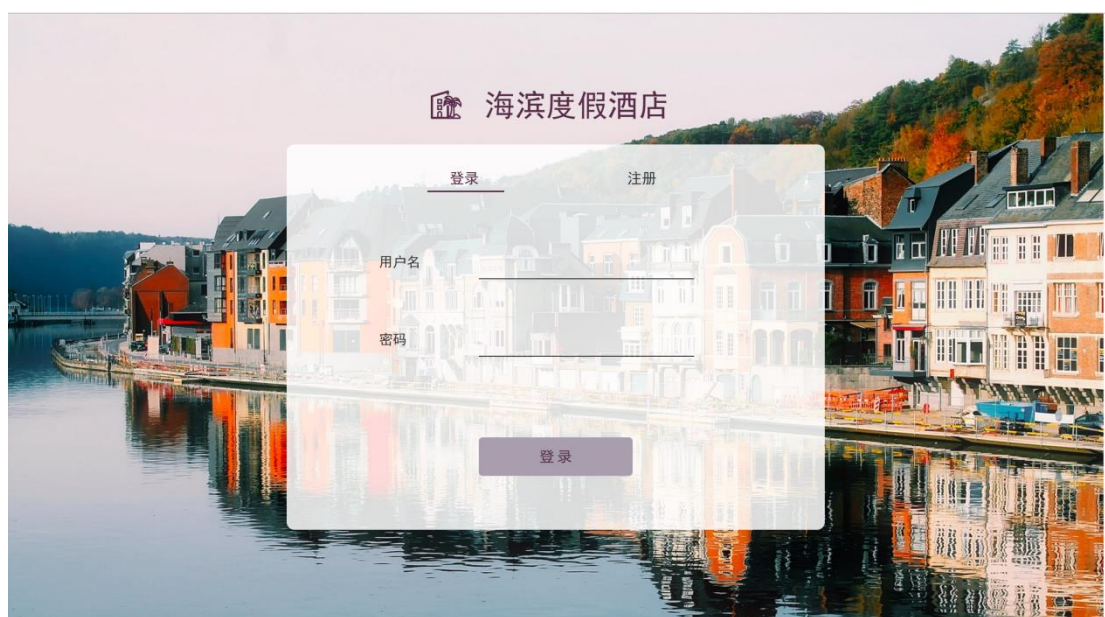
⑤前台安排服务事务界面



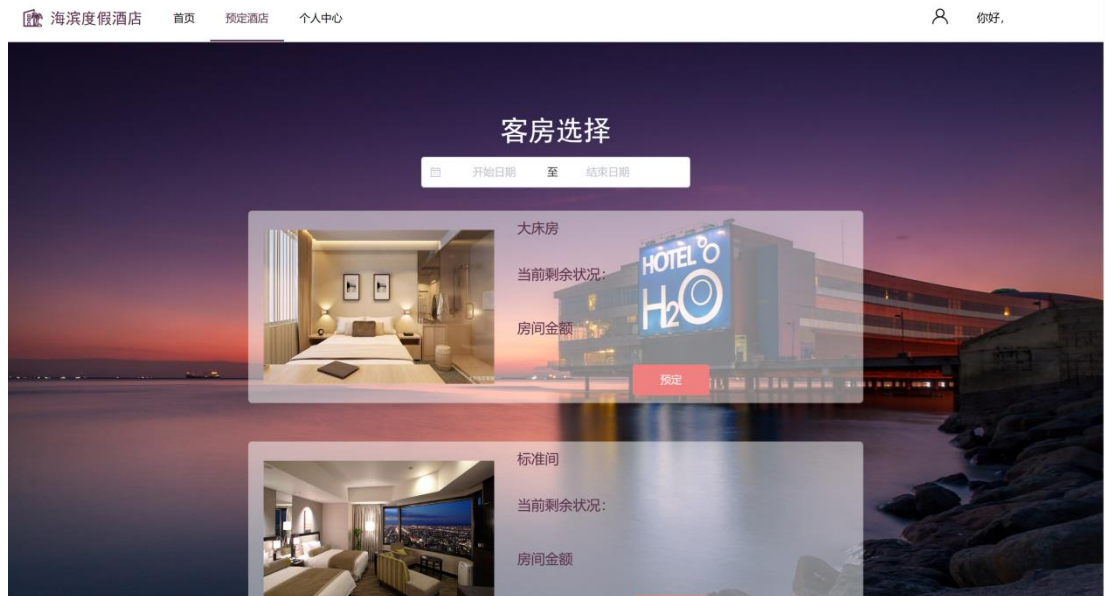
⑥订单查询修改界面



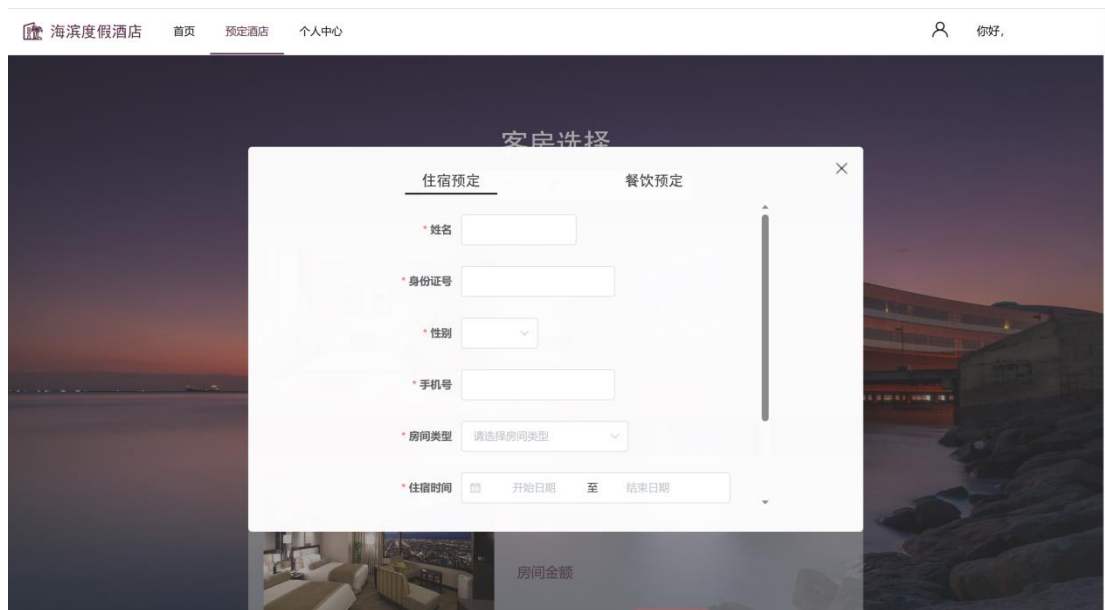
⑦客户登录/注册界面



⑧客户选择客房界面



⑨客户预订客房/餐饮界面



3.2 模块间接口设计

①员工登陆验证与各业务子系统之间的接口

描述：由于每种类型管理员登陆后进入的主界面不同，需要在主界面启动时传递管理员信息。

实现：构造方法的参数传递

例：财务管理子系统管理员登陆

```

public AffairManageForm(int id,string name,string type){
//财务管理主界面的构造方法，传递员工编号、姓名以及类型
this.id=id;
this.name=name;
this.type=type;
}

```

②可用客房查看模块与顾客提供客房要求之间的接口

描述：在管理员查看系统中可用客房时，必须根据顾客提供的信息进行检索（提供参数）。

实现：可用客房查看方法的参数传递

这是一个 DataGridView¹⁹

例：入住办理模块中的可用客房查看函数

```

List<> showUsableRoom(string type,DateTime start,Datetime end){
//三个参数分别为顾客提供的房间类型、入住时间和离开时间
List<Room> L = new List<>();
foreach room:
if(usable(room[i])){
if room[i] is usable from start to end(search each record
in liverecord and bookrecord)//遍历住宿记录和预约记录
L.add(room[i]);
}
return L;
}

```

③续住和换房模块也可以调用②中提到的 showUsableRoom 方法，参数与②中方法的形参形式一致，此处不再赘述。

④ 查询员工与删除/修改员工

描述：删除/修改某一个菜品需要查询或点击一个特定的菜品。此处需要接口传递菜品的编号。

实现：成员函数 clickedId 用来存放被点击的菜品的 id

```

private int clickedid=0;//若不点击默认为 0
public void onClick(object sender,...){
//监听点击事件的方法

```

```

this.clickedid=i.id;//i 是被点击的菜品对象，是数据库操作得到的
}
public void deleteDish() {
//删除数据库中对应的菜品
delete from Database where id=clickedid;
}

```

⑥账务查询接口

描述：对于财务管理员，账务查询是一个较为复杂的功能。在查询时，首先有多种账务类型的查询——住宿收入、餐厅收入、工资支

出和采购支出，其次，对于每种账务类型都有不同的查询条件。比如 20

餐厅收入中可以查询不同时间段、不同花费甚至是不同菜品的收入情

况。因此需要利用重载函数的多态性对不同查询做出不同响应。

实现：查询的重载函数

例：工资支出

```

public void query(string type,DateTime day) {
//查询某一天的工资支出
this.dataGridView.query(type, day);
}
public void query(string type, int workerId) {
//查询某一个员工的工资支出
this.dataGridView.query(type, workerId);
}

```

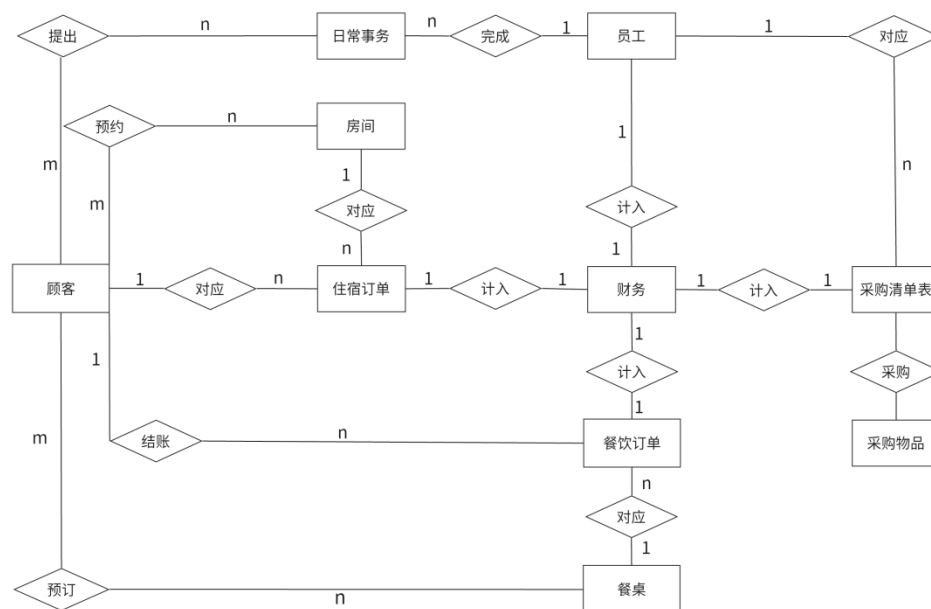
3.3 应用服务器与数据库通信接口

①mysql-connector-j-8.0.32.jar，内含 Java 程序与 MySQL 数据库的通信协议与方法，位于 External Libraries 文件夹下

②mybatis-3.5.11.jar，支持自定义 SQL、存储过程以及高级映射。MyBatis 可以通过简单的 XML 或注解来配置和映射原始类型、接口和 Java POJO 为数据库中的记录。

四：数据设计

4.1 ER 关系图



4.2 关系模式的建立

实体表

1. 财务表 (财务编号, 订单表号, 事务类型, 金额, 金额类型, 员工编号)
2. 采购清单表 (采购清单编号, 员工编号, 采购时间, 采购总金额)
3. 餐饮订单表 (餐饮订单编号, 顾客编号, 金额, 点餐时间, 餐桌表)
4. 餐桌表 (桌号人数, 是否空闲)
5. 房间表 (房间号, 正常房间, 价格, 会员房间价格, 房间类型, 是否空闲, 房间描述)

6. 顾客表 (顾客编号, 账号, 身份证号, 顾客类型, 密码, 顾客等级, 姓名, 卡余额, 电话)
7. 日常事务表 (事务编号, 事务内容, 顾客编号)
8. 物资表 (物资编号, 物资名称, 物资类型, 物资价格, 到期时间)
9. 员工表 (员工编号, 员工姓名, 员工身份证号, 员工地址, 员工类型, 具体工资, 是否空闲)
10. 住宿订单表 (订单编号, 顾客编号, 房间号, 总金额, 入住时间, 到期时间, 押金, 是否退房)

多对多关系

11. 预定房间表 (房间号, 顾客编号, 预定时间)
12. 采购表 (物资编号, 采购清单编号, 采购数目)
13. 顾客事务 (事务编号, 顾客编号, 事务内容)
14. 预定餐桌表 (桌号, 顾客编号, 预定时间)

4.3 数据库中表的实现

财务表	酒店的财务信息	
列名	数据类型	约束
财务编号 fianance_id	int	Primary key
订单编号 order_id	Int	
事务类型 business_type	Varchar(20)	business_type varchar(20) check(business_type = '餐饮订单收入' or business_type = '住宿订单收入' or business_type = '员工共工资支出' or business_type = '采购支出')
金额 money_type	Varchar(4)	
员工编号 employee_id	int	
财务事务时间 finance_time	date	

房间表		
列名	数据类型	约束
房间号 room_id	int	Primary key
房间类型 room_type	varchar(10)	check(room_type = '单人间' or room_type = '大床房' or room_type = '豪华间' or room_type = '三人间' or room_type = '双人间')

)
正常房间价格 room_price	Double	
会员房间价格 Room_vprice	double	
房间描述 room_describe	Varchar(40)	
是否空闲 room_status	Varchar(2)	check(room_status = '是'or room_status = '否')

住宿订单表		
列名	数据类型	约束
订单编号 aorder_id	Int	primary key
金额 money	Double	
入住时间 Checkin_date	Datetime	
到期时间 Checkout_date	Datetime	
押金 earnest_money	double	
是否退房 checkout	varchar(2)	check(checkout = '是'or checkout = '否')
房间号 room_id	Int	foreign key (room_id) references tb_room(room_id)
顾客编号 Customer_id	Int	foreign key (customer_id) references tb_customer(customer_id)

顾客表		
列名	数据类型	约束
顾客类型 Customer_type	Varchar(10)	
顾客编号 customer_id	int	Primary key
顾客账号 Customer_account	Int	
顾客密码 Customer_pwd	Varchar(40)	check(length(customer_pwd)>6)
顾客等级 Customer_level	Int	
顾客姓名 Customer_name	Varchar(10)	

身份证号 Customer_idcardnum	Varchar(36)	check(customer_idcardnum like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9X]')
顾客电话 Customer_telephone	long	check(length(customer_telephone) <= 11)
卡余额	double	

员工表		
列名	数据类型	约束
员工姓名 Employee_name	Varchar(10)	
是否空闲 Employee_status	Varchar(2)	check(employee_status = '是' or employee_status = '否')
员工类型 Employee_type	Varchar(20)	check(employee_type = '人事管理员' or employee_type = '财务管理员' or employee_type = '前台管理员' or employee_type = '采购人员' or employee_type = '后台管理员' or employee_type = '服务员')
具体工资 Employee_salary	Double	
员工编号 Employee_id	Int	Primary key
员工住址 Employee_address	Varchar(60)	
员工电话 Employee_telephone	Long	check(length(employee_telephone) <= 11)
员工身份证号 Employee_idcardnum	Varchar(36)	check(employee_idcardnum like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9X]')

采购清单表		
列名	数据类型	约束
采购时间 Purchase_time	Datetime	
采购总金额 Purchase_money	Double	
采购清单编号 Purchase_id	Int	Primary key
员工编号 Employee_id	int	foreign key (employee_id) references tb_employee(employee_id)

物资表		
列名	数据类型	约束
物资编号 Material_id	Int	Primary key

物资名称 Material_name	Varchar(20)	
物资类型 Material_type	Varchar(20)	
物资价格 Material_price	Double	
物资数量 Material_num	Int	

酒店事务表		
列名	数据类型	约束
房间号 room_id	int	foreign key (room_id) references tb_room(room_id)
是否完成 Business_type	Varchar(2)	check(business_status = '是'or business_status = '否')
完成员工编号 Employee_id	Int	foreign key (employee_id) references tb_employee(employee_id)
事务发起时间 Business_time	Datetime	
事务编号 Business_id	Int	Primary key

餐桌号		
列名	数据类型	约束
桌号 dtable_id	Int	Primary key
就餐人数 people_num	Int	
是否空闲 dtable_status	Varchar(2)	check(dtable_status = '是'or dtable_status = '否')

餐饮订单表		
列名	数据类型	约束
餐桌号 dtable_id	int	foreign key (dtable_id) references tb_dtable(dtable_id)
顾客编号 customer_id	Int	foreign key (customer_id) references tb_customer(customer_id)
金额 dorder_money	Double	
点餐时间 Dinner_time	Datetime	
订单编号 dorder_id	int	Primary key

预定房间表		
列名	数据类型	约束
顾客编号 customer_id	int	foreign key (customer_id) references tb_customer(customer_id)
预定时间 Bookroom_time	Datetime	
房间号 Room_id	int	foreign key (room_id) references tb_room(room_id)

采购表		
列名	数据类型	约束
物资编号 material_id	Int	foreign key (material_id) references tb_material(material_id)
采购清单编号 Purchase_id	int	foreign key (purchase_id) references tb_purchase(purchase_id)
采购数目 Purchase_num	int	

顾客事务细节表		
列名	数据类型	约束
事务编号 business_id	Int	foreign key (business_id) references tb_business(business_id)
事务内容 business_content	Varchar(40)	
顾客编号 customer_id	int	foreign key (customer_id) references tb_customer(customer_id)

预订餐桌表		
列名	数据类型	约束
桌号 dtable_id	Int	foreign key (dtable_id) references tb_dinnertable(dtable_id)
预订时间 book_time	Datetime	
顾客编号 customer_id	Int	foreign key (customer_id) references tb_customer(customer_id)

4.4 建表语句 (MySQL)

#财务表

```
create table tb_finance(
finance_id int primary key, #财务编号
finance_time date, #财务事务时间
```

```

money_type varchar(4), #金额类型
order_id int, #订单编号
employee_id int, #员工编号
business_type varchar(20)
check(business_type = '餐饮订单收入' or business_type = '住宿订单收入'
or business_type = '员工共工资支出' or business_type = '采购支出') #事务类型
);

```

#房间表

```

create table tb_room(
room_id int primary key, #房间号
room_type varchar(10) check(room_type = '单人间' or room_type = '大床房'
or room_type = '豪华间' or room_type = '三人间' or room_type = '双人间'), #房间类型
room_price double,
room_vprice double,
room_status varchar(2) check(room_status = '是' or room_status = '否'),
room_describe varchar(40)
);

```

#住宿订单表

```

create table tb_aorder(
room_id int,
customer_id int,
aorder_id int primary key,
money double,
checkin_date datetime,
checkout_date datetime,
earnest_money double, #押金
checkout varchar(2) check(checkout = '是' or checkout = '否'), #是否退房
foreign key (room_id) references tb_room(room_id),
foreign key (customer_id) references tb_customer(customer_id)
);

```

#顾客表

```

create table tb_customer(
customer_id int primary key,
customer_pwd varchar(40) check(length(customer_pwd)>6),
customer_level int,
customer_name varchar(10),
customer_idcardnum varchar(36) check(customer_idcardnum like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9X]'), #身份证号
customer_cardbalance double,
customer_telephone long check(length(customer_telephone) <= 11),
customer_account int,
customer_type varchar(10)
);

```

#员工表

```

create table tb_employee(
employee_id int primary key,
employee_name varchar(10),
employee_idcardnum varchar(36) check(employee_idcardnum like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9X]'),
employee_status varchar(2) check(employee_status = '是' or employee_status = '否'), #是否空闲
employee_type varchar(20) check(employee_type = '人事管理员' or employee_type = '财务管理员' or employee_type = '前台管理员' or
employee_type = '采购人员' or employee_type = '后台管理员' or employee_type = '服务员'),
employee_salary double,
employee_address varchar(60),
employee_telephone long check(length(employee_telephone) <= 11)
);

```

#采购清单表

```

create table tb_purchase(
employee_id int,
purchase_id int primary key,
purchase_time datetime,
purchase_money double ,
foreign key (employee_id) references tb_employee(employee_id)
);

```

#物资表

```

create table tb_material( #物资表*增加了一个物资数量
material_name varchar(20),
material_type varchar(20),
material_id int primary key,
material_price double,
material_num int
);

```

#酒店事务表

```

create table tb_business( #酒店事务表
business_id int primary key,
business_time datetime,
room_id int,
employee_id int,
business_status varchar(2) check(business_status = '是' or business_status = '否'),
foreign key (room_id) references tb_room(room_id),
foreign key (employee_id) references tb_employee(employee_id)
);

```

#餐桌表

```

create table tb_dinnertable(
dtable_id int primary key,
people_num int,

```

```
dtable_status varchar(2) check(dtable_status = '是' or dtable_status = '否')
);
```

#餐饮订单表

```
create table tb_dorder(
dorder_id int primary key,
customer_id int,
dorder_money double,
dinner_time datetime,
dtable_id int,
foreign key (customer_id) references tb_customer(customer_id),
foreign key (dtable_id) references tb_dtable(dtable_id)
);
```

#预定房间关系表

```
create table tb_bookroom(
bookroom_time datetime,
customer_id int,
room_id int,
foreign key (customer_id) references tb_customer(customer_id),
foreign key (room_id) references tb_room(room_id)
);
```

#采购关系表

```
create table tb_buymaterial(
purchase_id int,
material_id int,
purchase_num int,
foreign key (purchase_id) references tb_purchase(purchase_id),
foreign key (material_id) references tb_material(material_id)
);
```

#事务顾客对应关系表

```
create table tb_businessdetail(
business_id int,
customer_id int,
business_content varchar(40),
foreign key (customer_id) references tb_customer(customer_id),
foreign key (business_id) references tb_business(business_id)
);
```

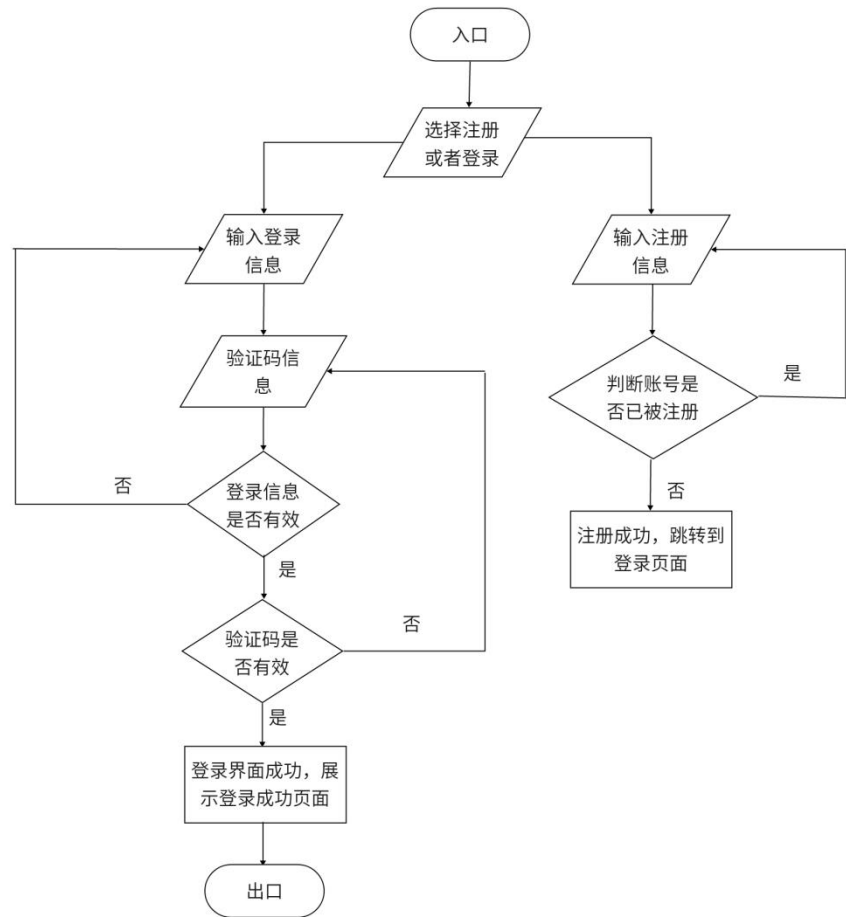
#预定餐桌表

```
create table tb_booktable(
customer_id int,
dtable_id int,
book_time datetime,
foreign key (customer_id) references tb_customer(customer_id),
foreign key (dtable_id) references tb_dinnertable(dtable_id)
);
```

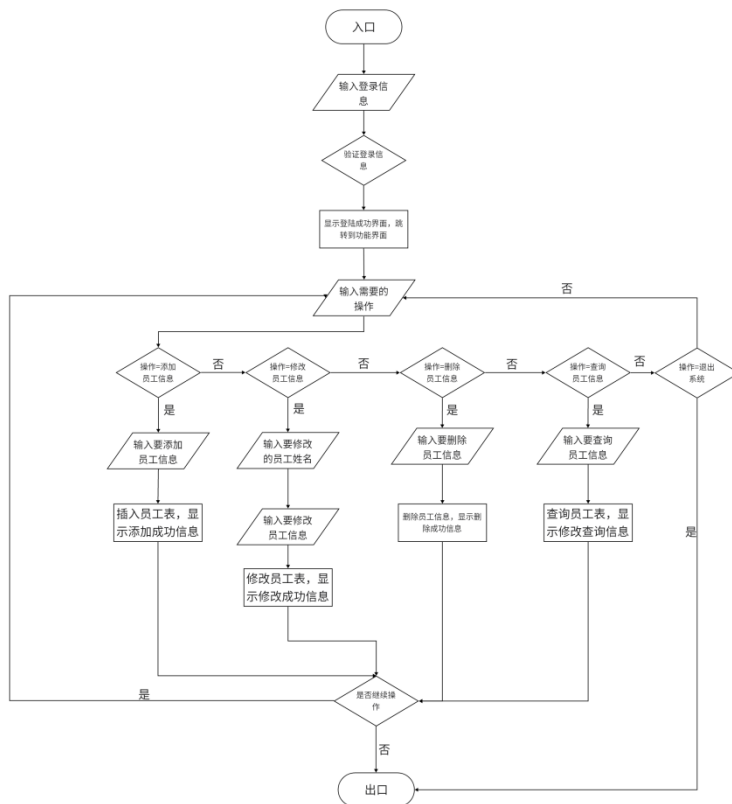
五：过程设计

5.1 主要功能模块的程序流程图

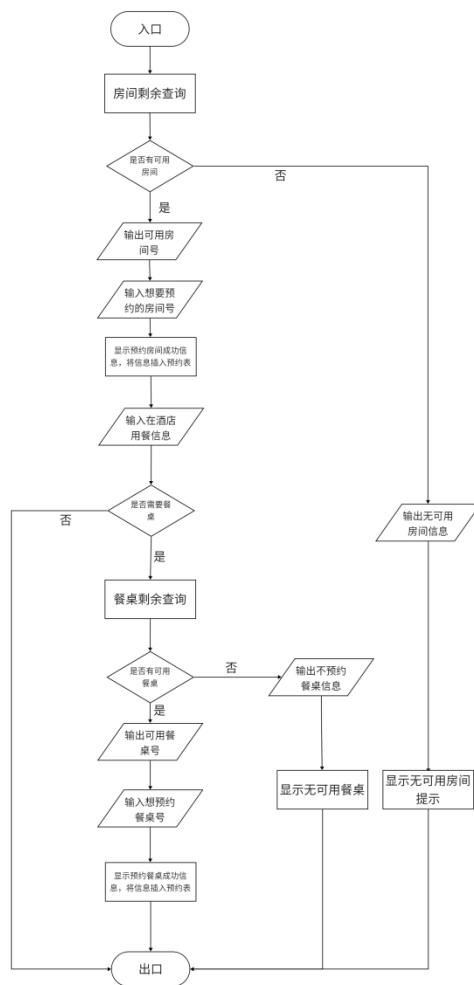
① 登陆验证



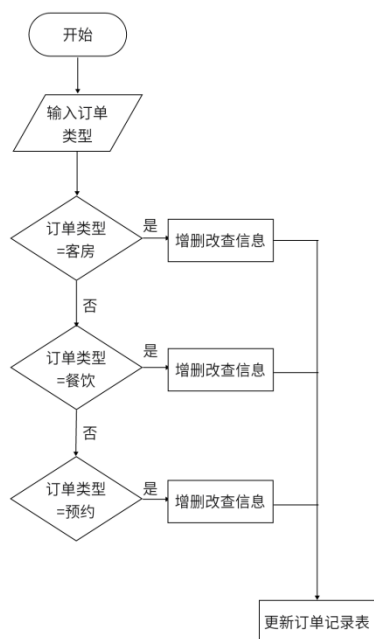
② 人事管理员管理人员



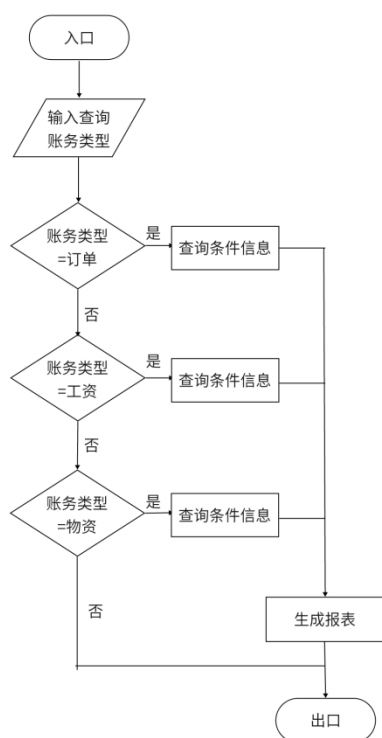
③ 酒店客房预约



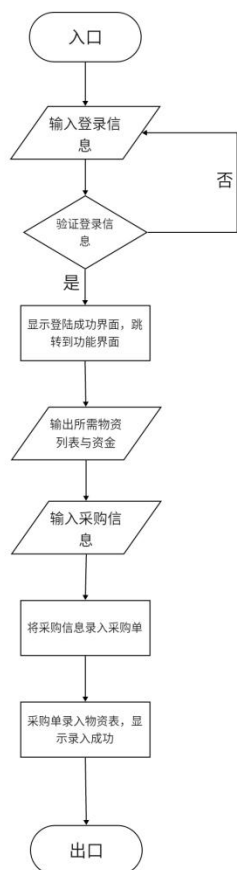
④ 订单处理



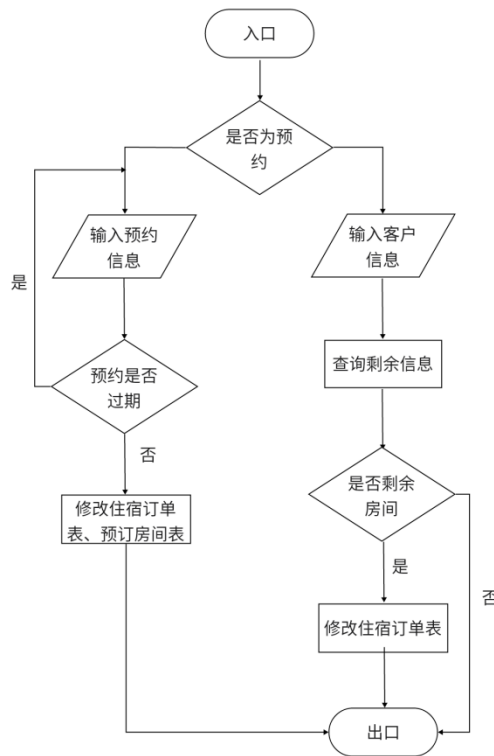
⑤ 财务查询



⑥ 采购



⑦入住办理



⑧后台管理员管理房间和餐厅信息

