# 小型数据库应用系统设计与开发

## 4.1系统设计目标

网上销售系统要求提供包括商品信息管理、查询、订购、销售等功能的网上交易平台，对客户和商店管理员应提供不同的操作界面和使用权限。

## 4.2 需求分析

### 4.2.1 功能需求

 商品信息管理：商店可以对商品信息进行管理，包括商品的类别、名称、描述信息、售价、图片、折扣等。

 客户信息管理：客户可以自助注册并管理自己的个人信息。

 商品查询：客户可以通过多种方式查询并且挑选网上商店出售的商品，通过在网上填写并确认订单的方式来购买商品。

 订单查询：客户可以管理自己的订单信息，查询订单的处理情况。

 订单处理：商店可以对客户的订单信息、汇款单信息进行审核，以确定是否发货，并修改订单状态。

### 4.2.2 性能需求

满足小规模的高效率的访问。

## 4.3 总体设计

采用B/S架构实现，系统功能模块设计如下：

网上销售系统

用户管理

商品管理

订单管理

登录

注册

上架

下架

购买

查询

添加

删除

修改

地址管理

图 4.1 系统功能设计图

项目功能结构图如下：

普通用户

商家管理用户

注册、管理个人信息

查询商品

下单

查询订单

添加商品

修改订单状态

查询订单

下架商品

注册、管理个人信息

图 4.2 系统结构功能设计图

## 4.4 数据库设计

系统的E-R设计图如下：

用户

订单

地址

商品

拥有

管理

对应

添加

1

n

n

m

1

n

1

n

对应

商品类型

1

n

id

姓名

昵称

类型

密码

id

地址

时间

状态

用户

商品

id

描述

商品存储

类型

名称

价格

图片

类型

id

用户

号码

类型邮编

地址

id

图 4.3 E-R 设计图

E-R设计图如图所示，每个用户对应多张订单，一张订单只对应一个用户，一个用户有多个收货地址，一个收货地址只对应一个用户，一个订单对应一个收货地址，一个收货地址对应多张订单，一个订单有多件商品，一件商品属于多个订单。

系统总共有四张表，分别是用户表，商品表，地址表，订单表，结果如下：

用户表（USER）

|  |  |  |
| --- | --- | --- |
| id | char | primary key |
| username | char | not null |
| password | char | not null |
| nickname | char | not null |
| type | int | not null |

表 4.1 用户表

订单表（ORDER）

|  |  |  |
| --- | --- | --- |
| id | char | primary key |
| date | Date | not null |
| status | int | default 0 |
| user\_id | char | foreign key |
| address\_id | char | foreign key |
| goods\_id | char | foreign key |

表 4.2 订单表

地址表（ADDRESS）

|  |  |  |
| --- | --- | --- |
| id | char | primary key |
| address | char | not null |
| tel | char | not null |
| post | char | not null |
| user\_id | char | foreign key |

表 4.3 地址表

商品表

|  |  |  |
| --- | --- | --- |
| id | char | not null |
| name | char | not null |
| price | double | not null |
| info | char |  |
| imag | char | default |
| store | int | default 0 |
| type\_id | char | foreign key |

表 4.4 商品表

## 4.5 详细设计与实现

本次实验采用B/S架构，前端提交请求，后台进行处理，然后响应前台的请求。前端通过JSP提交表单请求，后台通过Servlet进行请求处理，并返回结果。

各功能模块的具体实现如下：

**1.注册/登录模块功能实现**：

前端展示注册/登录界面，输入用户名和密码，选择普通用户登录，商店管理员用户登录，新用户注册三个按钮，提交请求，后台进行处理，流程图如下：

获取前端表单提交的请求

查询数据库是否有该用户

查询数据库密码是否正确

查询数据库获取用户类型

进入管理员后台界面

进入普通用户浏览商品展示界面

打印错误信息返回登录界面

打印错误信息返回登录界面

图 4.4 注册登录实现流程图

获取前端表格提交的请求

查询数据库是否有该用户

检查两次密码是否一致

将新用户插入数据库

注册成功，返回登录界面

用户已存在，注册失败

输入错误，注册失败

图 4.5 注册实现流程图

前端通过JSP动态页面表格提交请求，后台通过Servlet获取请求，并解析请求中的参数，然后进行数据库操作，具体为对user表的增删查改操作，并根据操作的结果返回响应的响应，前端获取响应，并跳转到响应的界面。

针对普通用户，在首页浏览商品界面，有修改个人信息的入口，可以维护个人信息，前端提交表格，后端Servlet进行响应的数据库操作，具体为更新操作。同时又查询个人订单的功能，会展示当前用户所有的订单。

**2.商品管理模块具体实现：**

商品管理包括两部分，前台针对普通用户是查询操作，用于展现所有的商品，同时还有根据商品类型进行查询，实现多种方式查询。在后台管理部分，商家管理员可以对商品进行增删查改操作。实现的具体流程为，前端提交请求，后台连接数据库，根据相应的请求执行对应的数据库语句，并返回响应，前端展示响应的界面，流程如下：

获取前端表格提交的请求

连接数据库

执行响应的SQL语句

返回响应的页面

图 4.6 商品管理实务流程图

**3.订单管理模块具体实现：**

普通用户在浏览商品界面，通过点击选中对应的商品，进入填写订单界面，在订单填写界面，用户确认自己购买的商品信息，并且填写地址，填写地址有两种方式，一种是选择已有地址，另外一种是添加一个新地址。流程图如下：

获取前端表格提交的请求

连接数据库

查询数据库是否为新地址

操作地址表，添加新地址

操作订单表，调价订单

返回首页

图 4.7 订单管理实现流程图

在商家管理员界面，对订单进行管理，查看已有的订单，并且修改当前新订单的状态，0表示未确认订单，修改为1表示确认订单，此处定义触发器一旦确认订单，就对商品表进行更新操作，更新商品的数量。

4.数据库部分操作实现：

user表，建表语句如下：

CREATE TABLE user(

user\_id INT(11) PRIMARY KEY AUTO\_INCREMENT,

username VARCHAR(100) NOT NULL ,

password VARCHAR(100) NOT NULL ,

nickname VARCHAR(100) UNIQUE NOT NULL ,

type int(5) NOT NULL

);

user\_id 为主码，并且自动增加

地址表，建表语句如下：

CREATE TABLE address(

addr\_id INT(11) PRIMARY KEY AUTO\_INCREMENT,

address VARCHAR(100) NOT NULL ,

tel CHAR(15) NOT NULL ,

post CHAR(10) NOT NULL ,

user\_id INT(11) NOT NULL ,

FOREIGN KEY(user\_id) REFERENCES user(user\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

addr\_id为主键，user\_id为外键，级联更新和删除。

订单标，建表语句如下：

CREATE TABLE ordert(

order\_id INT(11) PRIMARY KEY AUTO\_INCREMENT ,

date DATE NOT NULL ,

status int DEFAULT 0,

goodsname VARCHAR(100),

user\_id INT(11) NOT NULL ,

addr\_id INT(11) NOT NULL ,

FOREIGN KEY (user\_id) REFERENCES user(user\_id) ON DELETE CASCADE ON UPDATE CASCADE ,

FOREIGN KEY (addr\_id) REFERENCES address(addr\_id) ON DELETE CASCADE ON UPDATE CASCADE，

FOREIGN KEY (goods\_id) REFERENCES goodsgoods\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

order\_id为主键，user\_id,addr\_i，goods\_id为外键，级联更新和删除。

商品表，建表语句如下：

CREATE TABLE goods(

goods\_id INT(11) PRIMARY KEY AUTO\_INCREMENT,

goodsname VARCHAR(100) NOT NULL ,

price DOUBLE NOT NULL ,

info VARCHAR(200),

imag VARCHAR(100),

store INT DEFAULT 0,

type\_id INT(11),

FOREIGN KEY (type\_id) REFERENCES category(type\_id)

);

goods\_id为主键，type\_id为外键，级联更新和删除。

商品类型表，建表语句如下：

CREATE TABLE category(

type\_id int(11) PRIMARY KEY AUTO\_INCREMENT,

typesname VARCHAR(30) NOT NULL

);

type\_id为主键，自动增加。

在订单表上定义触发器，如下：

CREATE TRIGGER order\_count

AFTER UPDATE ON ordert

FOR EACH ROW

BEGIN

UPDATE goods SET store = store-1 WHERE goodsname = old.goodsname;

END;

## 4.6 系统测试

**实验环境：**操作系统：Ubuntu16.04 服务器：Apatch Tomcat M9.0 数据库：Mysql 浏览器：Firefox 集成开发环境：IDEA

**系统测试:** 启动Tomcat，然后部署Web项目，进行测试

**注册登录模块测试：**

在浏览器地址栏输入：<http://localhost:8080/jsp/login.jsp>进入注册登录界面，然后输入用户名和密码，进行登录，或者选择注册，注册新用户。登录注册页面如下：



图 4.8 注册登录界面

如图为注册登录界面，本次试验中已经有普通商家用户guojiani，密码：123456，以普通用户登陆后，进入到商品浏览首页，商家管理员用户:wanglin,密码：123456，登陆后进入管理后台页面.登录失败则返回响应的错误提示信息，并跳转到注册登录页面。

**商品浏览模块测试：**

普通用户登录后进入商品浏览界面，界面截图如下：



图 4.9 商品浏览界面

如图，首页展示了当前正在出售的商品，包括商品图片，商品名，商品价格，当前为全表查询，展示了所有商品，商品上面有一个查询窗口，可输入商品类型进行查询，此处为通过商品类型进行数据库查询操作，实现了所种方式进行数据库查询操作。

根据商品类型查询测试结果如下图所示：



图 4.10 按类型查询结果

如图所示，在输入框输入商品类型Summer，可得到对应的查询结果。

在首页左上角有维护个人信息的入口，点击进入个人中心界面，截图如下：



图 4.11 个人信息界面

如图所示为个人中心界面，展示了个人信息，包含用户名，密码等信息，还展示了当前用户已有的地址信息，还展示了当前用户已有的订单信息，有一条已经确认的订单信息。

当前用户对当前地址信息可进行修改和删除操作，点击修改操作，进入修改地址界面，截图如下：



图 4.12 修改地址信息界面

如图，填写需要修改的信息，对地址进行修改操作。

选择删除操作时，可将对应地址删除

**订单管理模块测试：**

在商品展示界面，用户点击对应的商品，选中，则进入订单填写界面，订单填写界面如图所示：

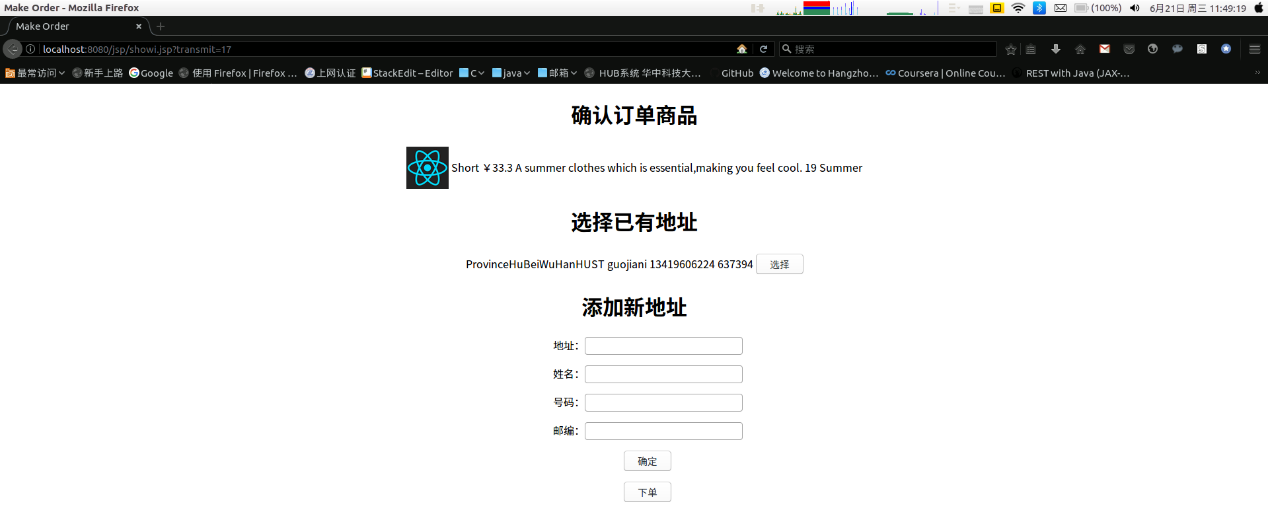


图 4.13 填写订单界面

如图所示为订单填写界面，展示了用户所选中的商品的详细信息，包含了对商品的描述信息，以及当前商品的库存量，下方展示了当前用户已有的地址信息，点击选择按钮，会选择当前的地址对下方地址栏填充，并且作为订单的地址字段加入订单，也可以在下方地址栏填写信息添加新地址，点击确定后，当前新地址会被加入到地址表中存储，对地址表进行更新操作。效果如图所示：

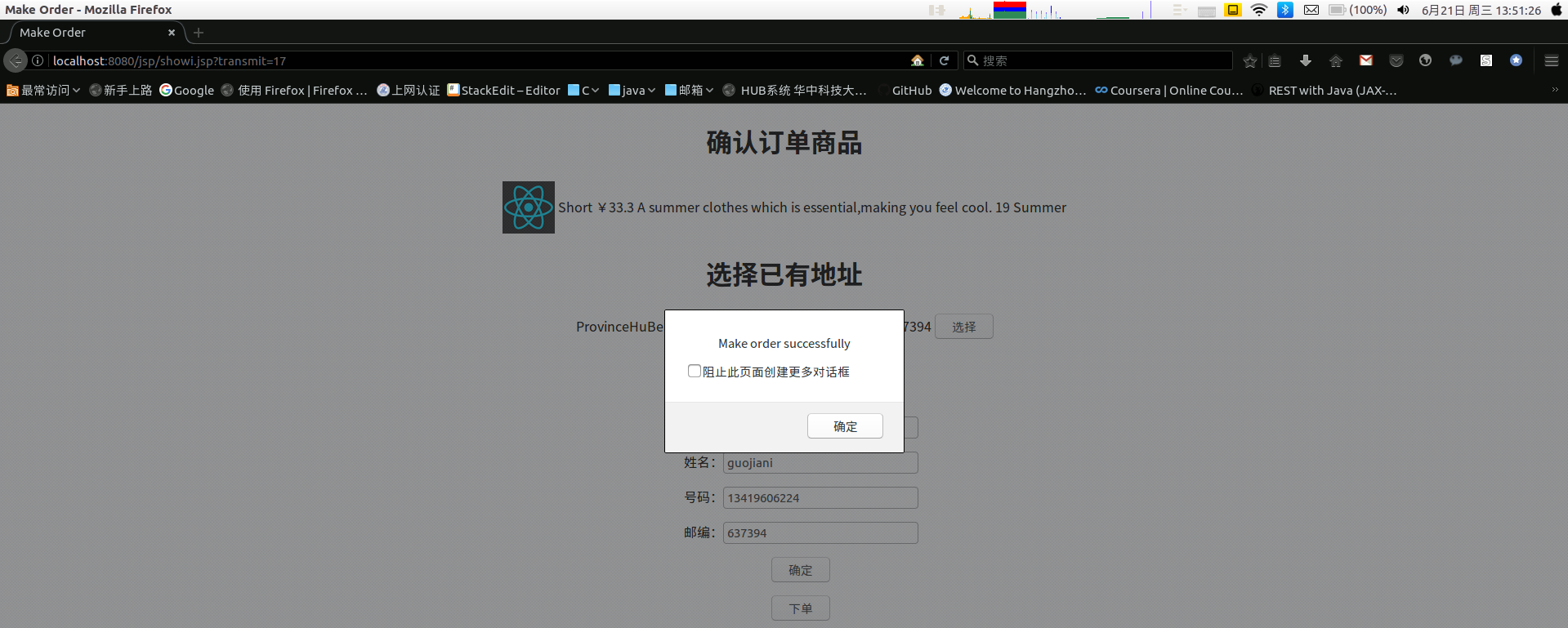


图 4.14 确认订单界面

如图，订单填写完毕后，点击确认按钮，弹出订单完成提示窗口，并返回首页。同时生成的订单会被插入到订单表中，对订单表进行插入操作。

在商店管理界面，后台可以对查看当前的订单信息，如图所示：

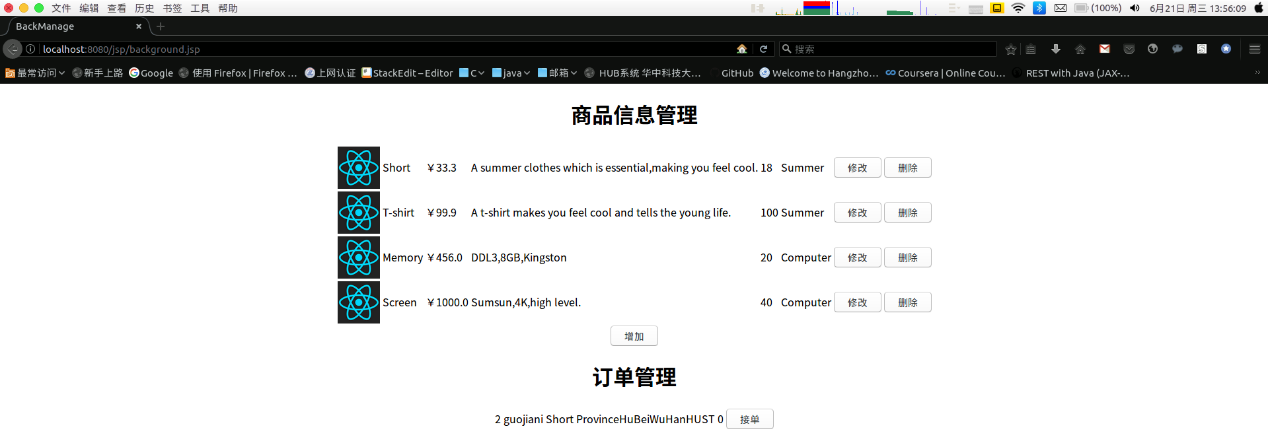


图 4.15 后台管理界面

如图所示，以商店管理员身份登录，查看后台数据，可以看到刚刚增加的订单信息，且当前订单状态为0，未被确认，上方对应商品库存为18,，商店管理员对订单进行确认，截图如下：

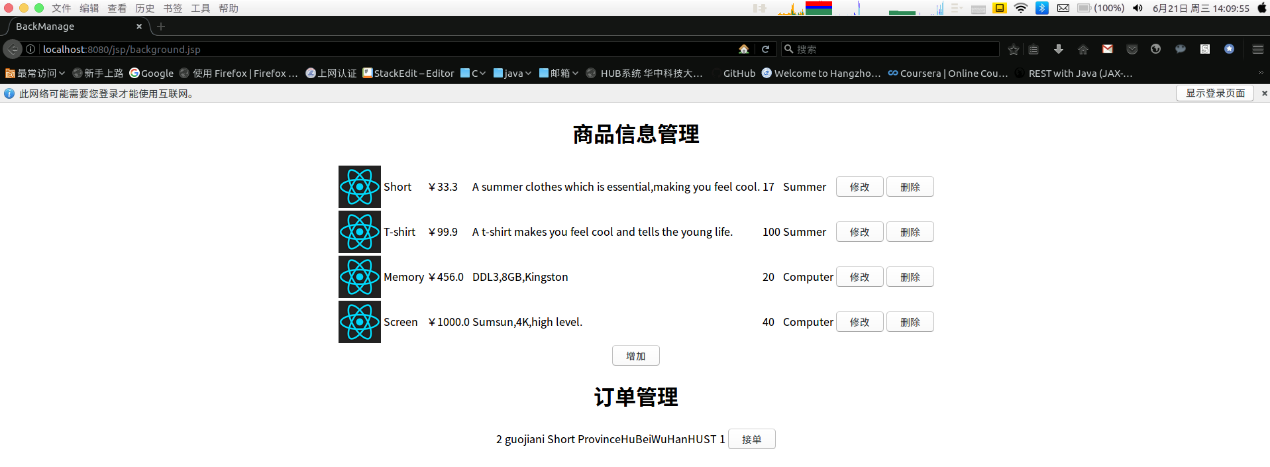


图 4.16 订单管理示意图

如图，商家点击接单按钮，订单状态由0变为1，并且上方商品库存量由18减少为17，这部分是通过设置触发器实现。

后台可对商品信息进行管理，包含增删查改操作，界面如图 4.16所示，在每一条商品信息后有修改和删除按钮，可以修改所选行的商品信息或者删除这一条商品，在下方有增加按钮，可以增加一条商品信息，截图如下：



图 4.17 修改商品信息

如图，选中对应的商品，可进行修改操作。

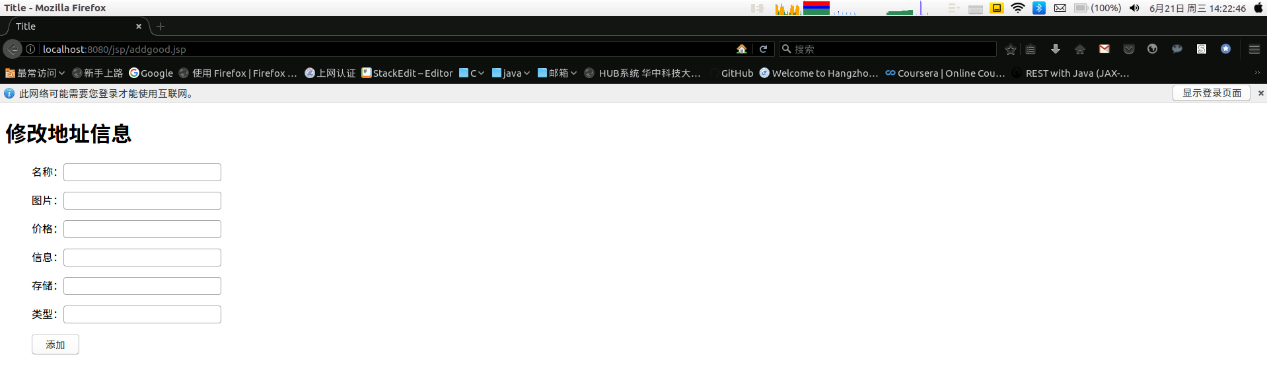


图 4.18 添加商品信息

如图，选择添加按钮，可添加一条商品信息。

综上，对系统的测试结束，在事务逻辑部分，首先对于普通用户，可以登录注册，然后可以管理个人信息，包括查看、修改、删除自己的用户信息、地址信息、订单信息；在商品展示页面，展示了所有的商品，用户可以查看所有的商品，也可以输入商品类型按照类型进行查询；选择对应的商品，进入订单填写界面，在订单填写界面，在订单填写界面，可以选择已有的地址，也可以添加一条新地址，确认后生成一条订单记录，记录被插入订单表，新地址被插入地址表。然后对于商店管理员用户，以管理员用户身份登录，进入后台管理界面，在后台管理界面，可以管理所有的商品信息，包括对商品信息的增删查改，对订单状态的修改。经过测试，上述功能完整实现。

## 4.7 系统设计与实现总结

分条目概括、总结应用系统设计的主要工作。

本次应用设计主要完成了一下工作：

1. 数据库部分的设计，包括数据字典，E-R图的设计，数据流图的设计，对数据依赖的设计。
2. 按照设计，建表，连接Mysql数据库，写建表语句，简历5张数据表。
3. 系统总体模块的设计，对各个功能模块进行设计
4. 采用B/S架构，对各模块的联系进行设计，包含功能的调用，会话的设计
5. 编写前端代码，绘制前端界面
6. 编写后端代码，和前端界面绑定相关事务逻辑
7. 调试，修正Bug