

《数据库系统原理》课程设计

系统实现报告

题目名称： 天下通物流管理系统

学号及姓名 13061193 刘乾

13061195 王文基

2015 年 12 月 27 日

目录

一、系统功能需求分析.....	3
1、系统概述.....	3
2、主要功能.....	3
二、系统功能结构设计.....	5
三、数据库基本表的定义.....	5
1.用户信息相关项.....	5
2、员工信息相关项.....	6
3、货物信息相关项.....	6
4、订单信息相关项.....	7
5、仓库信息相关项.....	7
6、管理员信息相关项.....	7
7、仓库明细表单信息相关项.....	8
8、已验订单信息表.....	8
9、已接货订单信息表.....	8
10、已入库订单信息表.....	9
11、已分发订单信息表.....	9
12、已完成订单信息表.....	9
四、触发器的定义与实现.....	10
1、check_trigger.....	10
2、receiv_trigger.....	10
3、enter_trigger.....	11
4、distribute_trigger.....	11
5、finish_trigger.....	12
五、存储过程的定义与实现.....	12
1、changeStatus.....	12
2、ProcAddMoney.....	13
3、ProcChangeAddr.....	13
4、ProcChangePhone.....	13
5、ProcChangeEmail.....	14
6、ProcFillMoney.....	14
六、系统实现结果.....	14
七、总结.....	23

一、系统功能需求分析

1、系统概述

天下通物流管理系统是逻辑清晰，功能明确的面向物流管理领域的数据库系统。该数据系统集成成了下单、接货、入库、派送等一系列物流管理流程，通过操纵订单、客户、仓库等实体对象达到了物流管理流程的一体化。面向管理员、用户、员工等相关人员，支持增、删、改、查等各种操作。

2、主要功能

本物流管理系统主要功能分为查询功能和信息的增删和修改功能，同时本系统面向客户和员工和管理员有三套界面模块，分别有不同的功能。

2.1、对于客户来说

- 缴费功能

客户可以通过自己的支付宝、网银等电子支付手段等向自己在本物流公司已经注册过的会员账号内充值。

- 下单寄件

客户可以通过寄件功能在网上通过帮助自助操作下单，实现网上下单操作。

- 取消未配送订单

对于还没有配送的订单，用户可以选择取消订单。

- 查询订单状态

客户还可以查询自己的货物当前的状态，共有几种状态：上门取货中，待入库，待配送，待收货，已经签收。

- **修改个人信息**

客户可以通过个人信息页修改自己的用户姓名，联系电话，通信地址与电子邮箱。

- **注册**

还没有在本物流公司申请会员账号的客户可以通过自助注册与手机验证

2.2、对于员工来说

- **关于接货人员—确认订单功能**

在客户在系统中成功下订单后，在所有接待人员的栏目条中会显示出需要进行接货的订单，此时接货人员要先将该订单设置为取货中状态，然后前往用户所在地址进行接货。当接货完毕后，接货人员将订单状态更改为待入库状态。如果接货人员上门取货失败，则可以将订单取消，此时订单将被丢弃到回收站。

- **关于入库管理员—货物入库功能**

在订单被更改为待入库状态时，订单会自动从接货人员的页面消失，而出现在入库管理员的页面中。入库管理员可以前往负责该订单的接货人员处接货，放入仓库。之后将订单的状态更改为待配送状态，并需要添加货物在仓库中的位置信息。

- **关于配送员—货物配送功能**

在订单状态被更改为被配送状态时，订单会自动出现在配送员的页面中。配送员根据订单中已经登记好的货物在仓库中的位置信息和收货地址等配送到收货地。

2.3、对于管理员来说

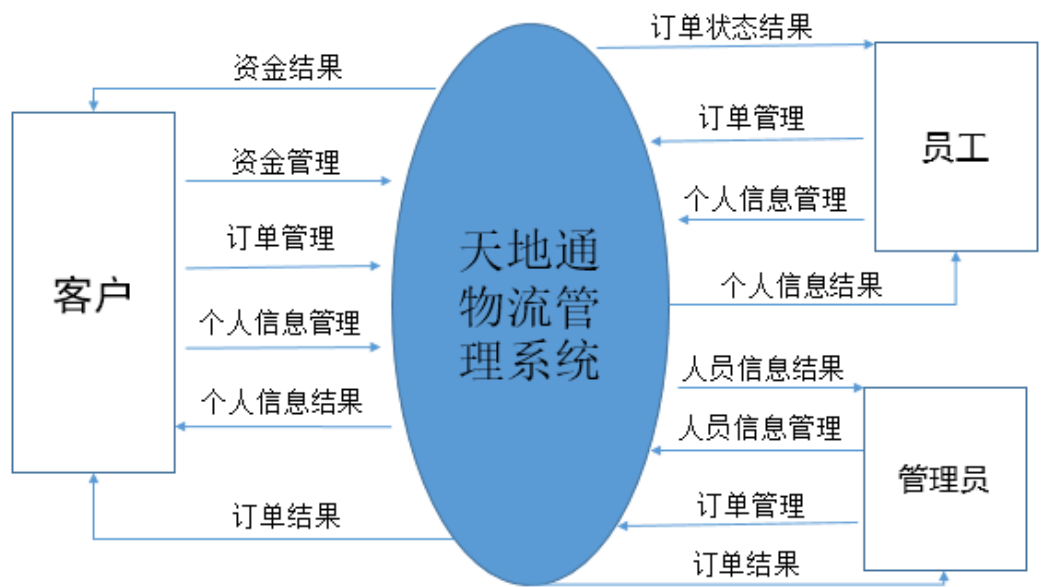
- **管理员工和客户**

管理员可以通过管理界面管理员工和客户的资料信息，其中包括了客户的密码，账号与账户内金额等。

- **管理订单**

管理员可以看到所有的订单的直接责任人与责任时间， 并且可以查询和变更订单。

二、系统功能结构设计



三、数据库基本表的定义

1. 用户信息相关项

数据项名	数据类型	长度	备注
user_id	varchar	20	用户账号
user_pwd	varchar	20	密码
user_name	varchar	5	用户姓名
user_email	varchar	30	用户邮箱
user_id_num	varchar	30	用户身份证号
user_addr	varchar	50	用户地址
user_phone	varchar	20	用户联系方式

user_account	float	10	用户账户余额
--------------	-------	----	--------

2、员工信息相关项

数据项名	数据类型	长度	备注
worker_pwd	varchar	20	密码
worker_birth	varchar	20	出生日期
worker_native	varchar	20	籍贯
worker_occupy	varchar	10	职务
worker_id	varchar	20	员工编号
worker_gender	varchar	5	性别
worker_name	varchar	5	姓名
worker_phone	varchar	20	联系电话
worker_admit	varchar	20	参加工作时间

3、货物信息相关项

数据项名	数据类型	长度	备注
cargo_type	varchar	10	种类
cargo_name	varchar	10	名称
cargo_id	varchar	20	货物编号
cargo_volume	float	10	体积

cargo_weight	float	10	重量
--------------	-------	----	----

4、订单信息相关项

数据项名	数据类型	长度	备注
cargo_id	varchar	20	货物编号
user_id	varchar	20	客户编号
receiver_id	varchar	20	收货人编号
money	float	10	金额
order_id	varchar	20	订单编号

5、仓库信息相关项

数据项名	数据类型	长度	备注
wareh_id	varchar	20	仓库编号
wareh_volume	float	10	仓库容量
wareh_remain	float	10	仓库余量

6、管理员信息相关项

数据项名	数据类型	长度	备注
admin_id	varchar	20	编号

admin_name	varchar	10	姓名
admin_pwd	varchar	20	密码
admin_accounts	varchar	10	账号

7、仓库明细表单信息相关项

数据项名	数据类型	长度	备注
wareh_id	varchar	20	仓库编号
row	int	2	行号
column	int	2	列号
cargo_id	varchar	20	货物编号

8、已验订单信息表

数据项名	数据类型	长度	备注
worker_id	varchar	20	员工编号
order_id	varchar	20	订单编号
check_time	datetime	19	验收时间

9、已接货订单信息表

数据项名	数据类型	长度	备注
------	------	----	----

worker_id	varchar	20	员工编号
order_id	varchar	20	订单编号
receiv_time	datetime	19	接货时间

10、已入库订单信息表

数据项名	数据类型	长度	备注
worker_id	varchar	20	员工编号
order_id	varchar	20	订单编号
enter_time	datetime	19	入库时间

11、已分发订单信息表

数据项名	数据类型	长度	备注
worker_id	varchar	20	员工编号
order_id	varchar	20	订单编号
distribute_time	datetime	19	分发时间

12、已完成订单信息表

数据项名	数据类型	长度	备注
worker_id	varchar	20	员工编号

order_id	varchar	20	订单编号
finish_time	datetime	19	完成时间

四、触发器的定义与实现

1、check_trigger

说明：设置触发器，在插入已确认订单列表之后被激活，将单号相同的订单列表中的订单的状态设置为已检单。

```
create trigger check_trigger
after insert on checkList
for each row
update orderList set state = "checked"
where order_id = new.order_id;
update orderlist set receiveCargo_time = localtimestamp() where order_id =
new.order_id;
```

2、receiv_trigger

说明：设置触发器，在插入已收货订单列表之后被激活，将单号相同的订单列表中的订单的状态设置为已收货，同时将订单号相同的已检单列表中的订单删除。

```
create trigger receiv_trigger
after insert on receivList
for each row
begin
update orderList set state = "received"
where order_id = new.order_id;
```

```

update orderlist set receiveCargo_time = localtimeStamp() where order_id =
new.order_id;
delete from checkList where order_id = new.order_id;
end

```

3、enter_trigger

说明：设置触发器，在插入已入库订单列表之后被激活，将单号相同的订单列表中的订单的状态设置为已入库，同时将订单号相同的已收货列表中的订单删除。

```

create trigger enter_trigger
after insert on enterList
for each row
begin
update orderList set state = "entered"
where order_id = new.order_id;
update orderlist set enterWarehouse_time = localtimeStamp() where order_id =
new.order_id;
delete from receivList where order_id = new.order_id;
end

```

4、distribute_trigger

说明：设置触发器，在插入已分发订单列表之后被激活，将单号相同的订单列表中的订单的状态设置为已分发，同时将订单号相同的已入库列表中的订单删除。

```

create trigger distribute_trigger
after insert on distributeList
for each row

```

```

begin
update orderList set state = "distributed"
where order_id = new.order_id;
update orderlist set distribute_time = localtimestamp() where order_id =
new.order_id;
delete from enterList where order_id = new.order_id;
end

```

5、finish_trigger

说明：设置触发器，在插入已完成订单列表之后被激活，将单号相同的订单列表中的订单的状态设置为已完成，同时将订单号相同的已分发列表中的订单删除。

```

create trigger finish_trigger
after insert on finishList
for each row
begin
update orderList set state = "finished"
where order_id = new.order_id;
update orderlist set finish_time = localtimestamp() where order_id = new.order_id;
delete from distributeList where order_id = new.order_id;
end

```

五、存储过程的定义与实现

1、changeStatus

说明：根据表单的 id 将表单的状态设置为 newState

```

create procedure changeStatus(id varchar(20),newState varchar(20))

```

```
begin
update orderList set state = newState where order_id = id ;
end
```

2、ProcAddMoney

说明：根据用户的 id 将用户的余额加上 add_money

```
create procedure addMoney(id varchar(20),add_money float)
begin
update client set user_account = user_account + add_money where user_id = id ;
end
```

3、ProcChangeAddr

说明：根据用户的 id 将用户的地址设置为 newAddr

```
create procedure changeAddr(id varchar(20),newAddr varchar(20))
begin
update client set user_addr = newAddr where user_id = id ;
end
```

4、ProcChangePhone

说明：根据用户的 id 将用户的电话设置为 newPhone

```
create procedure changePhone(id varchar(20),newPhone varchar(20))
begin
update client set user_phone = newPhone where user_id = id ;
end
```

5、ProcChangeEmail

说明：根据用户的 id 将用户的邮箱设置为 newEmail

```
create procedure changeEmail(id varchar(20),newEmail varchar(20))  
begin  
update client set user_email = newEmail where user_id = id ;  
end
```

6、ProcFillMoney

说明：根据订单的 id 将订单的金额设置为 money

```
create procedure fillMoney(id varchar(20),money float)  
begin  
update orderList set money_amount = money where order_id = id ;  
end
```

六、系统实现结果

登录界面



注册界面



The registration interface consists of a form and a modal dialog. The form has three input fields: '账号' (Account) with 'TestData', '密码' (Password) with '123456', and '姓名' (Name) with '王文基'. Below these is a '联系方式' (Contact Information) field with '13222323345'. A teal '注册' (Register) button is at the bottom. A teal modal dialog is open, displaying the title '注册' (Register), the message '您已成功注册，快来登录体验一下吧' (You have successfully registered, please log in to experience it), and two buttons: 'OK' and 'Cancel'.

账号

密码

姓名

联系方式

注册

注册

您已成功注册，快来登录体验一下吧

OK Cancel

登陆后的界面：



The user dashboard after login features a teal header with the text '欢迎回来，刘乾!' (Welcome back, Liu Qian!). Below the header is a navigation bar with three links: '订单管理' (Order Management), '个人信息' (Personal Information), and '个人管理' (Personal Management). The main content area contains a large icon of a document with a checklist. At the bottom, there are two buttons: an orange '新订单' (New Order) button and a teal '查询订单' (Query Order) button.

欢迎回来，刘乾!

订单管理 个人信息 个人管理

新订单 查询订单

（注：登陆后出现的用户界面，分成三个部分：订单管理、个人信息、个人管理，其中的订单管理界面有新订单和查询订单两个操作）

用户下单界面

订单

货物类型

货物体积

货物重量

货物名称

收货方账号

金额

确认

（注：点击新订单，出现如上界面，用户自行填写下单界面的自己订单的详细信息，点击确认显示订单金额，完成下单操作）

用户订单管理界面

订单管理

我下的订单

订单编号：9

点我查看订单详细信息

<<

人

人

人

人

人

>>

撤销订单

订单最近更新时间：2015/12/28 19:12:27

（注：点击查询订单，出现上述的界面，用于查看订单的状态和完成订单的撤销等工作。

我们的订单状态有接货、入库、分发、配送、送达状态，这些状态随着订单的流动，动态变化，用户可以在这个过程中实时查看自己的每一个订单的状态）

用户订单管理中查看订单详细信息界面



（注：点击右上角的点我查看订单详细信息会出现订单的详细信息）

用户订单管理界面撤销订单操作



（注：在没有接货之前可以撤销，等到订单处于已接货状态之后就不能再撤销了）

员工登录中的接货员界面



（注：这个是员工中的接货员登陆后的界面，接货员可以查询到最新的刚下的订单，刷新后出现待确认的订单）

接货员点击确认后的界面



(注：点击确认按键后出现，标注其为已确认状态)

查询已确认订单的结果

欢迎回来，接货员

待确认订单

已确认订单

	请求	订单确认人	订单编号	订单确认时间
▶	接货成功	接货员	9	2015/12/28 22:14

刷新

(注：出现接货成功按键，准备进入下一状态，刷新后出现待标记接货成功的订单)

点击接货成功按键

欢迎回来，接货员

待确认订单

已确认订单

请求

确认接货成功

OK

刷新

(注：提示接货成功)

员工中的入库员登录界面



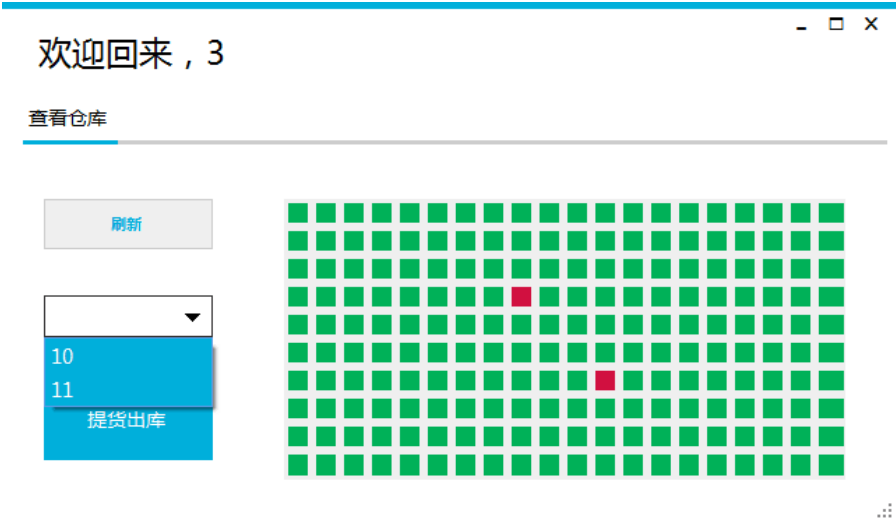
（注：出现入库按键，刷新后出现待入库的订单）

点击入库出现界面



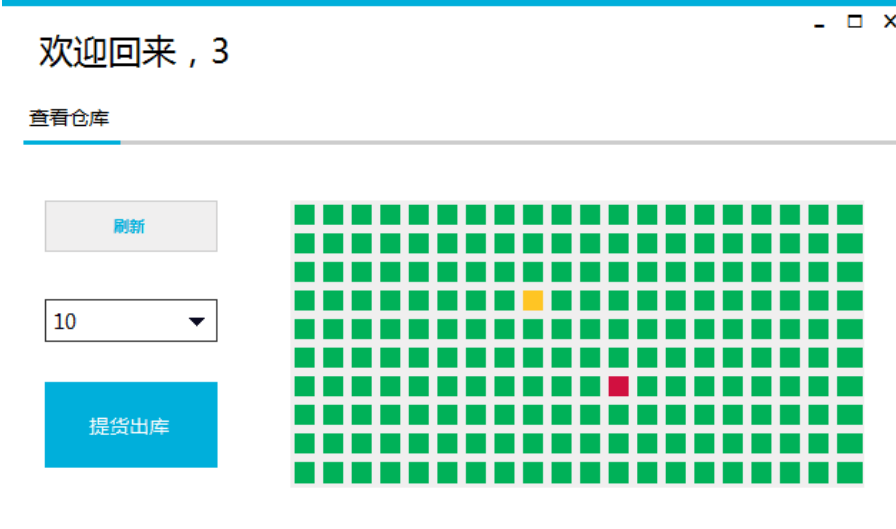
（注：该可视化的界面形象地展示了那些地方时可用的，就像电影选票一样，入库员可以选择货物待入的位置，选择后变为红色）

配送员查看仓库界面



(注：配送员查看可视界面得到待配送货物的地点)

取走 11 号货物之后的状态



点击上图的区域得到的货物信息

货物信息

货物名称

3232

货物类型

普通货物

货物重量

232

货物体积

321

确认

否

管理员登录界面

员工管理 用户管理 订单管理

员工编号

姓名

职务

请填入员工号

电话

查看

性别

籍贯

增加员工

工作时间

生日

（注：管理员登录界面包括员工管理，用户管理，订单管理三个部分）

查询员工界面

员工管理 用户管理 订单管理

员工编号

姓名

刘乾

职务

receiver

13

电话

18811771230

查看

性别

M

籍贯

shanxi

增加员工

工作时间

2015/2/4

生日

2015/2/4

...

查询用户界面

员工管理 用户管理 订单管理

客户名

siviltaram

姓名

刘乾

电话

18811771230

地址

232

邮箱

身份证号

142201199602190018

查看

账户余额

8

...

七、总结

王文基：

数据库基本操作的学习：

纸上得来终觉浅，绝知此事要躬行。课堂上的理论学习很酣畅，但是实际的操练更是不可避免，这是普遍真理，更是计算机专业的一个必要的要求。在这次的结对编程中我和我的同伴一起合作通力完成。我在这次实践中练习了数据库操作的基本语言，学习了基本的增删改查

的 MySQL 语句，从实战的层面上对于 SQL 语言有了更深入的理解。

数据库高级进阶：

老师在课堂上对于触发器和存储过程提及甚少，于是我们在实际使用的时候通过自学解决了这一问题，这些从 MySQL5.0 之后才开始支持的语言特性让我体会到了其方便之处，对于数据库的操作和编程有了更好的理解。

防御性编程：

在这次的结对编程中和我的队友交流的过程中我意识到了防御性编程的重要性，之前我在编程的时候比较注重主体功能的实现，而经常忽视那些异常和细节，尽管平时老师们强调了很多次防御性编程的重要性，在上学的面向对象编程中也着重练习了相关的技能，但是到了自己编程的时候还是有很多的问题没有考虑到，当时以为先把主体功能做了之后再管那些细节，但是事实证明并没有那么多的时间和精力，我的队友则不然他有很好的防御性编程的习惯，这一点在我们合作的过程中我感受良多，也是触动很大，使我有决心在以后的编程实战中注意培养这个好习惯。

刘乾：

本次数据库课设让我感受良多。

首先是关于美化的控件的选取，一个漂亮的界面能给用户最直观的美的感受，这也是一个面向客户的好的软件的基本要求。我们在刚开始数据库做应用时，虽然投入了很多精力在后端数据库增删改查的使用，但是在前端的页面美化，布局设计，包括每个按钮的位置，配色，系统的交互，库管员和配送员能看到的那些带了特效的界面，都是花了很多心思设计出来的。而为了做到美化并且可自定义修改这一点，我们使用了 MetroFramework 控件来实现，并且将它也推荐给了一些同样使用 C# 做数据库课设的同学。

第二点感受颇深的就是关于防御性编程。在刚起步的时候，很多时候 `MySqlDataReader` 都会忘记 `close` 而导致异常，到后面还有包括触发器只能在代码里创建一次等各种大大小小的问题。在克服这些以后，在和队友写的代码进行整合与对接的时候，发现了一个严重性问题：队友对防御性编程的思想并不是很注重，所以一旦出现错误或非法的情况，队友不会主动去处理。比如在注册的时候没有检查账号是否有冲突就直接插入数据库中，比如有些特殊的数据是需要特殊的格式，比如日期，这种队友直接使用了

TextBox并且没有使用正则判断其格式直接就让它可以插入数据库，最后导致代码很多地方都有很多可能会产生严重异常的漏洞。在经过和队友的一番紧张整合和修改后，最终才将代码的防御性增强了一些，至少比之前好很多。这一点我也深有感触：在涉及到数据库编程时，一定要有良好的防御性编程的思想，否则很容易产生插入异常等。

第三点感受比较深的就是关于我们设计的系统的理想和现实的差距的问题。我们在设计报告中即使考虑到了一些冗余的数据，但是有些东西在不实现时真的没有办法体会到其是否真的是“冗余”数据。而且数据库的应用的设计上，复杂性比想像更大。我们一开始进行了很多的思考与设计，但是最终还是讲原来想实现的一些功能砍掉来保证了最终其他部分效果的完善。有些时候不是因为时间的原因，而是因为自己使用的技术与能力比较受限。总之，数据库的设计的理想与现实还是存在一定差距的。

我认为数据库课设本身很有意义，它让我体验了如何能够从设计阶段开始，设计并实现一个数据库应用的全过程。并且跟队友的结对编程也让我感受良多，我从他那里学到了很多，他也从我这里学到了很多好的思想与习惯。希望以后能够再参与一些类似项目，从设计到开发到完善，能够实践自己学到的理论。