

Usage Guide

Zheng (Godric) Gu

400047153

A. Components

1. Client

GUI is written using MFC (Microsoft Foundation Classes)

MFC is a class library provided by Microsoft. It encapsulates Windows Api in the form of C++ classes. The classes inside contain a lot of windows handles classes and windows built-in control tools classes.

2. Development tool

The Cpp and h files and all the resource files should be included in a MFC project ,compiled and bulilt using visual studio 2010.

3. Database management system

The database management system this book shop management system connect to is Sql server 2008 R2. To configure the database, use the bookshop_Data and bookshop_Log files contained in the folder and put them under

C:\Program Files\Microsoft SQL Server\MSSQL10_50.MSSQLSERVER\MSSQL\Data

Then connect this bookshop database in sql server 2008 R2.

B. How to use this system

First, add supplier information into supply1, supply2, supply3 tables. Configure user information in OperalInfor table. Login to the system

through login interface and enter the user interface corresponding to your user's authority.

manager users can access:

1. system setting module: A manager can add new users (manager or clerk) to this system, exit this system and change a user and re-login.

2. stock management module: (1) Check the stock records. (2) Select the lowest price among different suppliers.

3. sale statistics module: (1) check all the sale records (2) check the ranking list of the books. The ranking could be done through different methods.

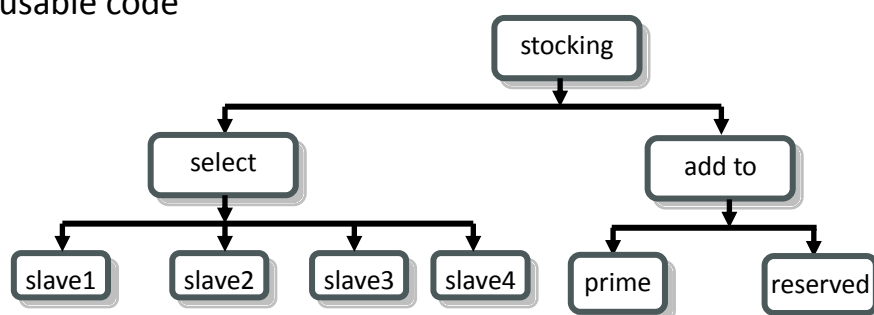
clerk users can access:

1. system setting module: (1) Exit this system (2) change a user and re-login.

2. Refund module: (1) check if the book sold out is from this bookshop. (2) refund to storage.

3. Vending function module: (1) show the storage from prime database or reserved database. (2) book inquiry (3) select the book and calculate the total price.

C. Reusable code



Now, the first 3 slaves read supplier records from different databases. If we have more databases to access or if we want to get access to every big supplier's database respectively, we can have more slaves to submit the supplier records. In the CStockManage class, we have 3 different methods below corresponding to 3 different slaves.

```
void CStockManage::GetDataFromSlave1()  
void CStockManage::GetDataFromSlave2()  
void CStockManage::GetDataFromSlave3()
```

So, if we have more slaves like this and submit records to the master module. We can reuse this kind of structure.

Also, if we have more databases to use as reserved databases. When make update, or insertion to the prime database, we also need to make all the databases consistent, So that if the prime database can't be accessed, we can still get records from other backup databases.