Payment Sub-System

Shiyun Liu

1. System Introduction

This system is a small online payment sub-system. In traditional shopping system, users always use a third-party system to pay for their bills. They add money into their accounts and pay for the ordered items. It's more convenient and useful for the customers who always make shopping online.

2. Development Environment

Operating System: Windows 7
Development Software: MyEclipse 10
Server: Apache Tomcat 7.0
Database: MySQL 6.0
Technologies: JSP + JavaBean

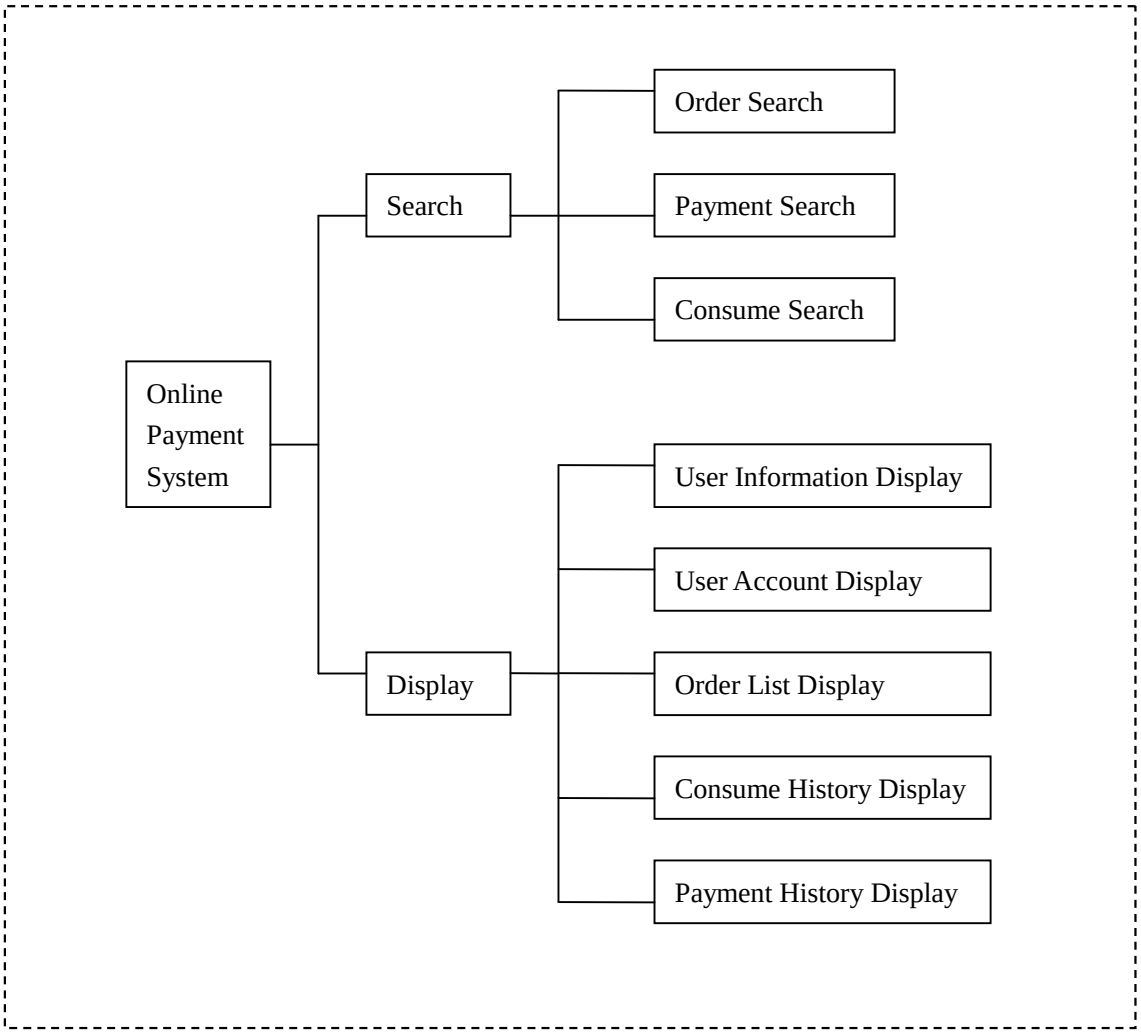
3. Functional Analysis

The whole payment system is based on the communications with database. Users make operations on the data in the database. If a user wants to buy an item and he/she is also a member of the payment system, he can use this payment system account to pay for his/her bills. After finishing payment, a record will be added into his/her account. But before paying, he/she also needs to add enough money or E-coin into his/her account.

This small system realize search and display functions. Users can view the information of orders, payment and consume records. And they can search the information they want by ID, Type and Time.

3.1 Function Module

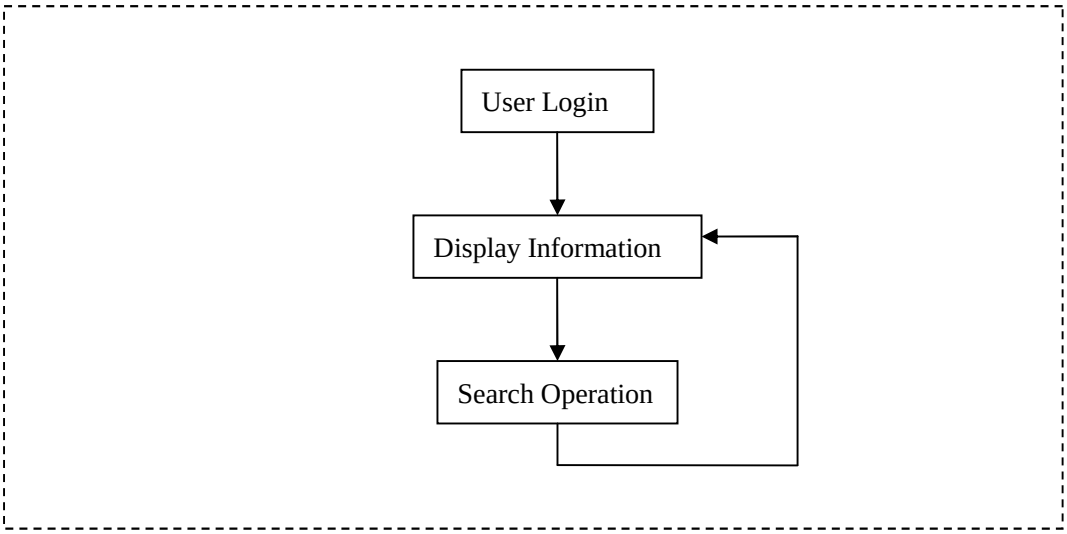
This online payment system can roughly divided into two modules: Search and Display.



Each of the two module contains their sub modules. Search module contains order search, payment search and consume search and Display contains user information display, user account display, order list display, consume history display and payment history display.

3.2 System Flow Diagram:

Only members can use this system, so that the flow of this system is only based on members.



Users login the payment system and the system displays the user information and user account information. Then users can search by ID, Type or Time and the system will display the results

4. Database Design

4.1 Tables:

The database of this system contains four tables: USER, ADD, CONSUME, ORDER.

USER:

Name	Type	NULL	Default	Extras
Keys (1)				
Primary Key	ID			unique
Fields (6)				
ID	int(11)	No		
NAME	varchar(20)	Yes	<NULL>	
PASSWORD	varchar(10)	Yes	<NULL>	
COIN	int(11)	Yes	<NULL>	
CONSUME	int(11)	Yes	<NULL>	
TYPE	varchar(20)	Yes	<NULL>	

ADD:

Name	Type	NULL	Default	Extras	C..
Keys (1)					
Primary Key	ADDID			unique	
Fields (5)					
ADDID	int(11)	No			
TYPE	varchar(20)	Yes	<NULL>		
COIN	int(11)	Yes	<NULL>		
DATETIME	datetime	Yes	<NULL>		
USERID	int(11)	Yes	<NULL>		

CONSUME:

Name	Type	NULL	Default	Extras	C..
Keys (1)					
Primary Key	CONSUMEID			unique	
Fields (5)					
CONSUMEID	int(11)	No			
PRICE	int(11)	Yes	<NULL>		
DATETIME	datetime	Yes	<NULL>		
USERID	int(11)	Yes	<NULL>		
PRODUCTNAME	varchar(50)	Yes	<NULL>		

ORDER:

Name	Type	NULL	Default	Extras	C..
Keys (1)					
Primary Key	ORDERID			unique	
Fields (6)					
ORDERID	int(11)	No			
TYPE	varchar(20)	Yes	<NULL>		
PRICE	int(11)	Yes	<NULL>		
DATETIME	datetime	Yes	<NULL>		
USERID	int(11)	Yes	<NULL>		
PRODUCTNAME	varchar(20)	Yes	<NULL>		

4.2 Data:

Add data into the four tables so that the system can display that.

USER:

ID	NAME	PASSWO...	COIN	CONSUME	TYPE
1	ABC	111111	100	250	VIP
2	bcd	111111	20	120	COM

ADD:

ADDID	TYPE	COIN	DATETIME	USERID
1	Bank	15	2006-01-01 00:00:00	1
2	Check	30	2006-02-01 00:00:00	1
3	Online	15	2006-03-01 00:00:00	1
4	Online	100	2006-04-01 00:00:00	1
5	Agent	50	2006-05-01 00:00:00	1

CONSUME:

CONSUME...	PRICE	DATETIME	USERID	PRODUCTNA...
1	2	2005-01-01 12:00:00	1	Apple
2	25	2006-02-03 00:00:00	1	Pink Skirt
3	34	2006-03-05 00:00:00	1	T-Shirt
4	2	2006-03-10 13:05:00	1	Apple
5	221	2006-03-13 00:00:00	1	Bed

ORDER:

ORDERID	TYPE	PRICE	DATETIME	USERID	PRODUCTNA...
1	Processing	15	2006-04-04 00:00:00	1	
2	Successfully	24	2006-03-05 00:00:00	1	
3	Failed	55	2006-02-01 00:00:00	1	
4	Successfully	5	2006-03-01 00:00:00	1	
5	Successfully	54	2006-01-01 00:00:00	1	

4.3 Database Connection:

(1) Config.properties:

```
conn_type=
conn_url=jdbc:mysql://localhost:3306/buy?useUnicode=true&characterEn
         coding=UTF-8
conn_user=root
conn_pwd=yuyouji
conn_driver=com.mysql.jdbc.Driver
```

(2) Config.java:

```
public class Config {
  private static Properties prop = new Properties();
  static {
      try {
          prop.load(Config.class
                  .getResourceAsStream("config.properties"));
```

```
        } catch (IOException e) {
            System.out.println("File:config.properties no find,PLS check
out!");
            e.printStackTrace();
        }
    }
    public static String CONNECTION_TYPE = prop.getProperty("conn_type");
    public static String CONNECTION_URL = prop.getProperty("conn_url");
    public static String CONNECTION_USER = prop.getProperty("conn_user");
    public static String CONNECTION_PWD = prop.getProperty("conn_pwd");
    public static String CONNECTION_DRIVER = prop.getProperty("conn_driver");

   }
```

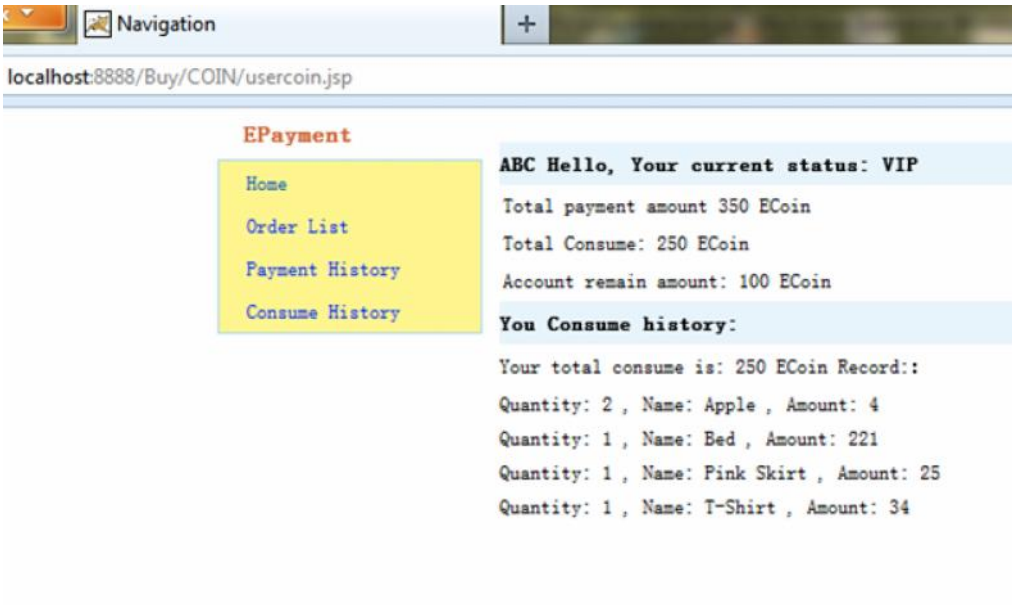
(3) DBConnect.java:

```
public class DBConnect {

    public static Connection getConnection() {
        Connection conn = null;
            try {

                Class.forName(Config.CONNECTION_DRIVER).newInstance();

                conn = DriverManager.getConnection(Config.CONNECTION_URL,
                        Config.CONNECTION_USER, Config.CONNECTION_PWD);
            } catch (Exception e) {
                e.printStackTrace();
            }
        if (conn == null) {
            System.out.println("no get connection!throws Exception");
        }
        return conn;
    }
   }
```

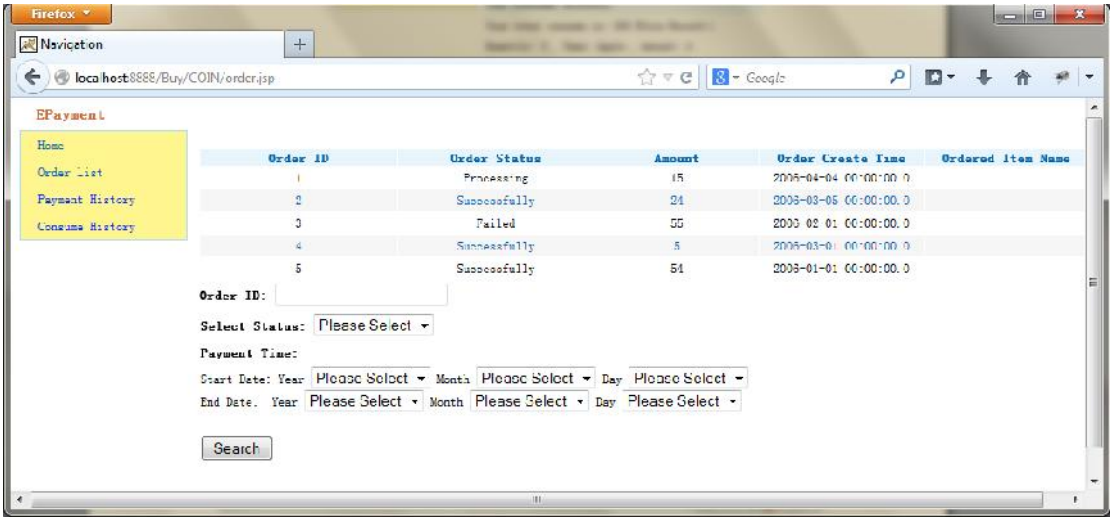
5. Result

(1) Home Website:

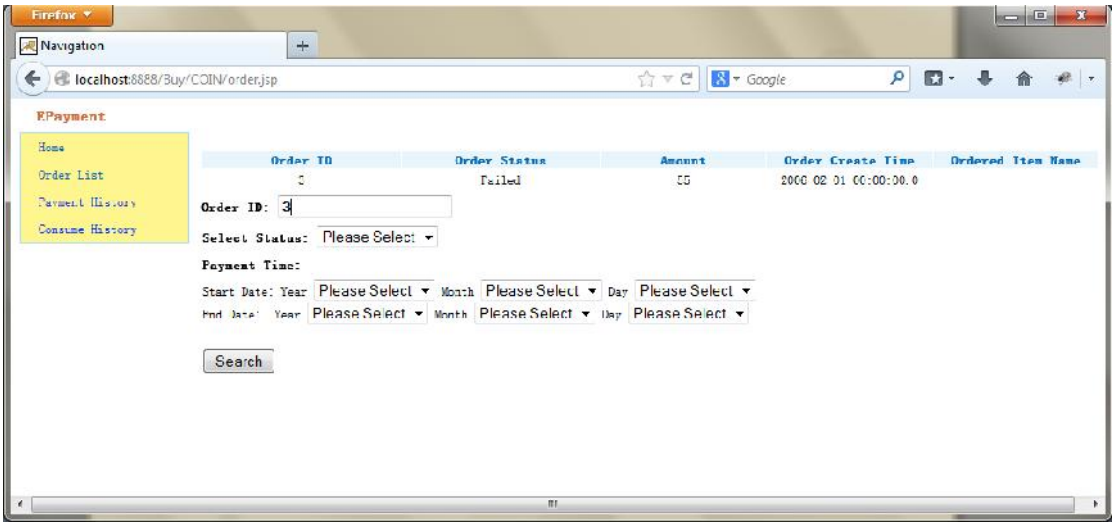
After Login, the system display the user information and user's account information.



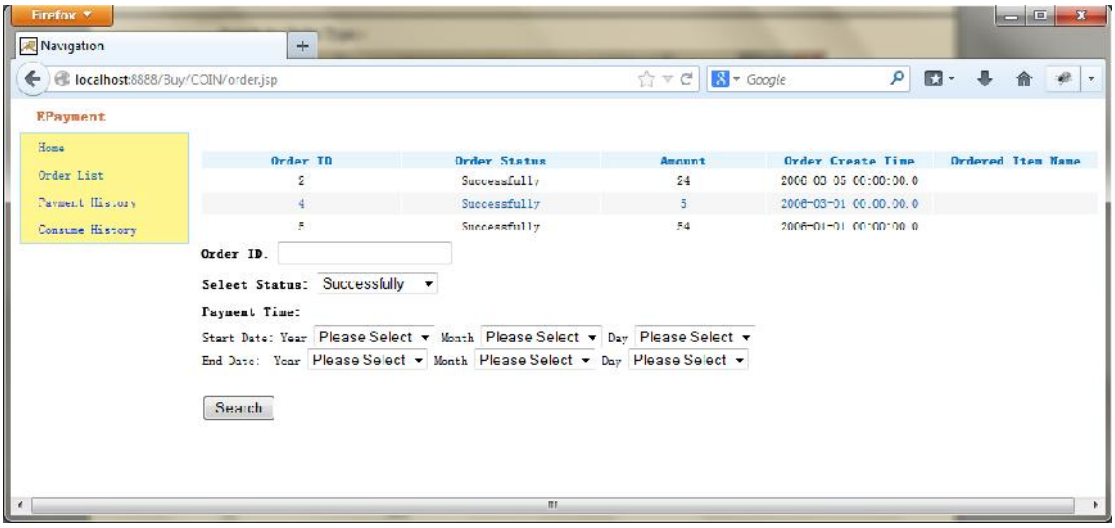
(2) Order List:



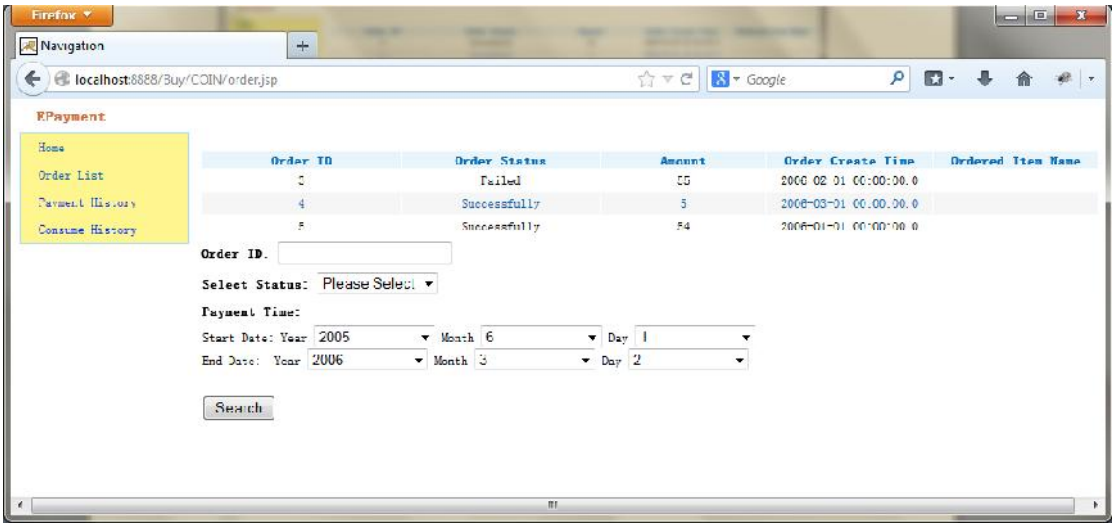
Search by Order ID:



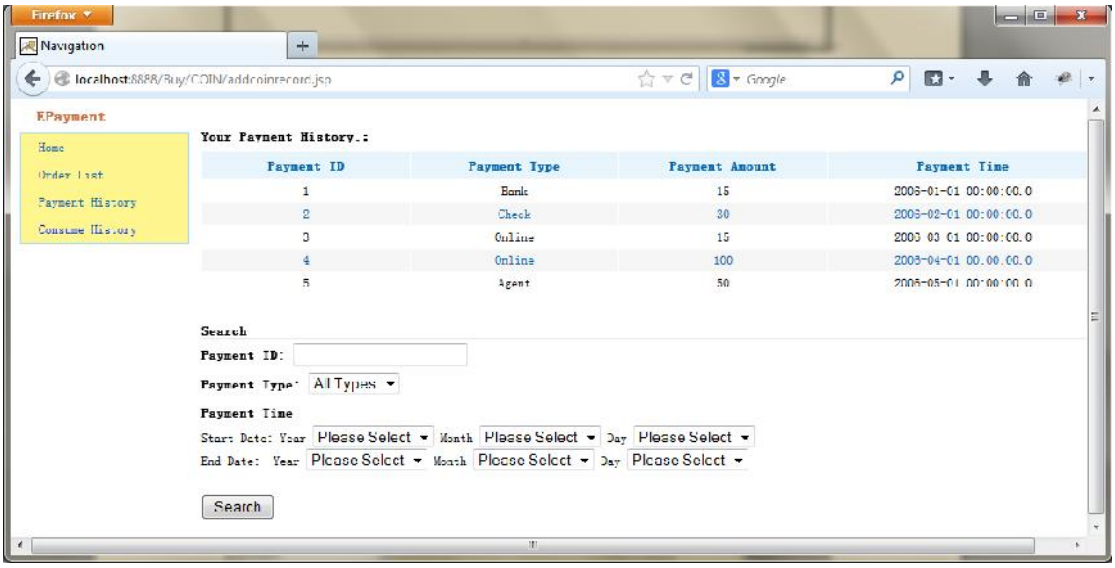
Search by Order Status:



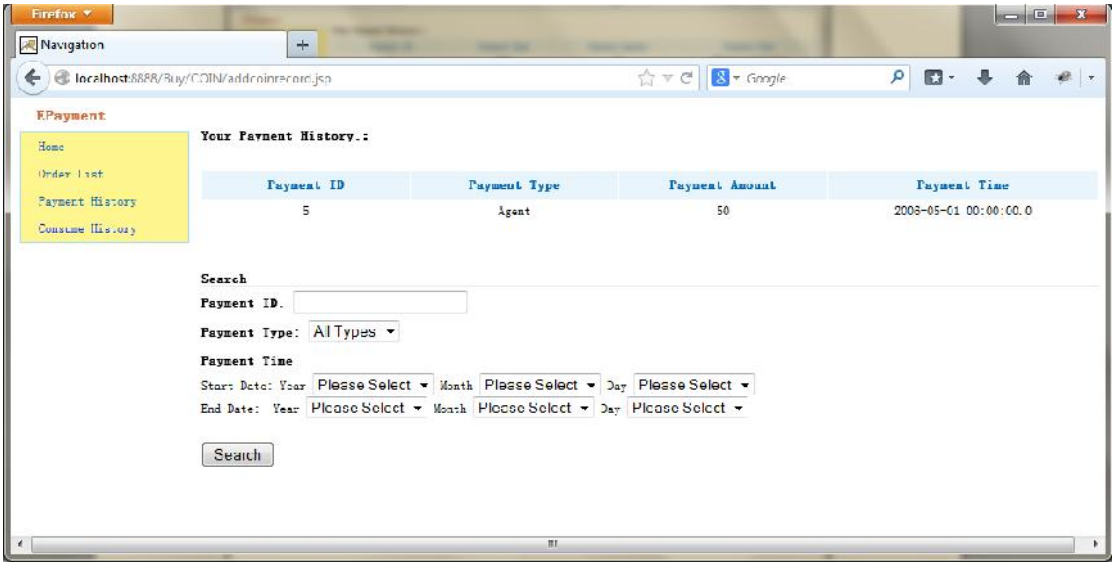
Search by Order Creation Time:



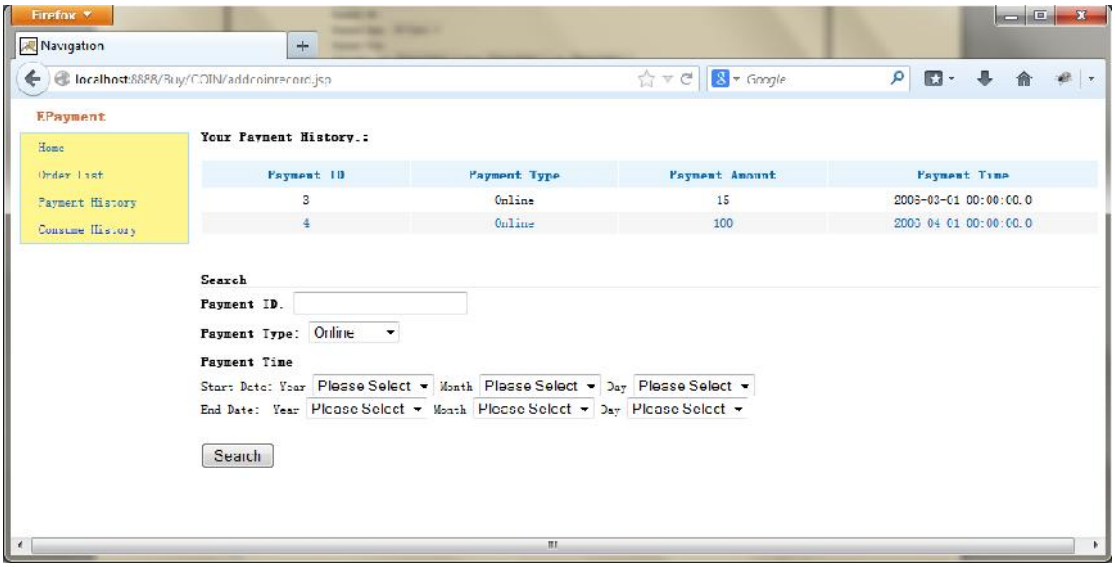
(3) Payment History:



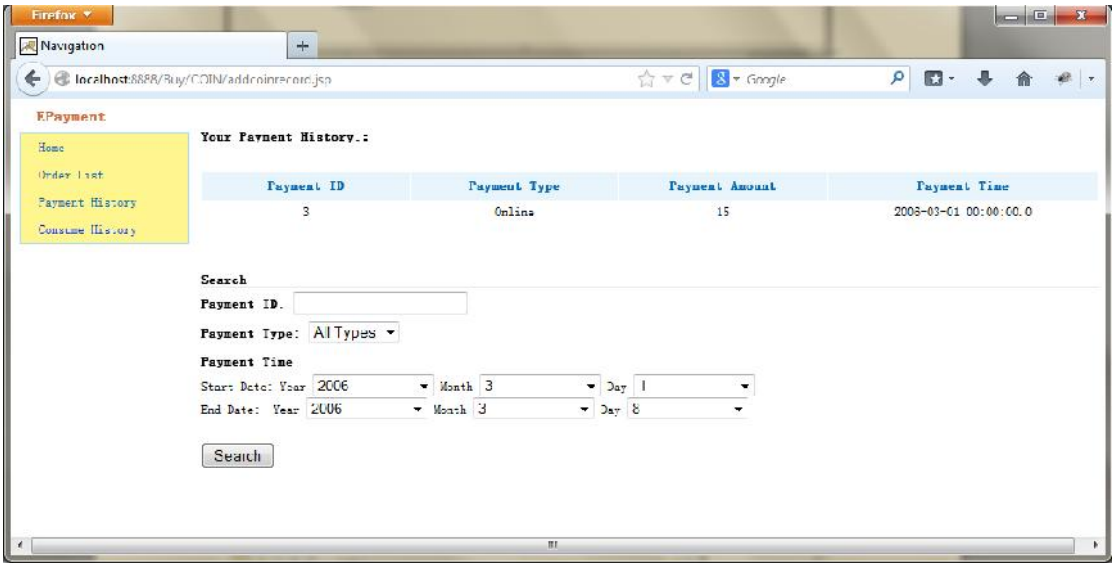
Search by Payment ID



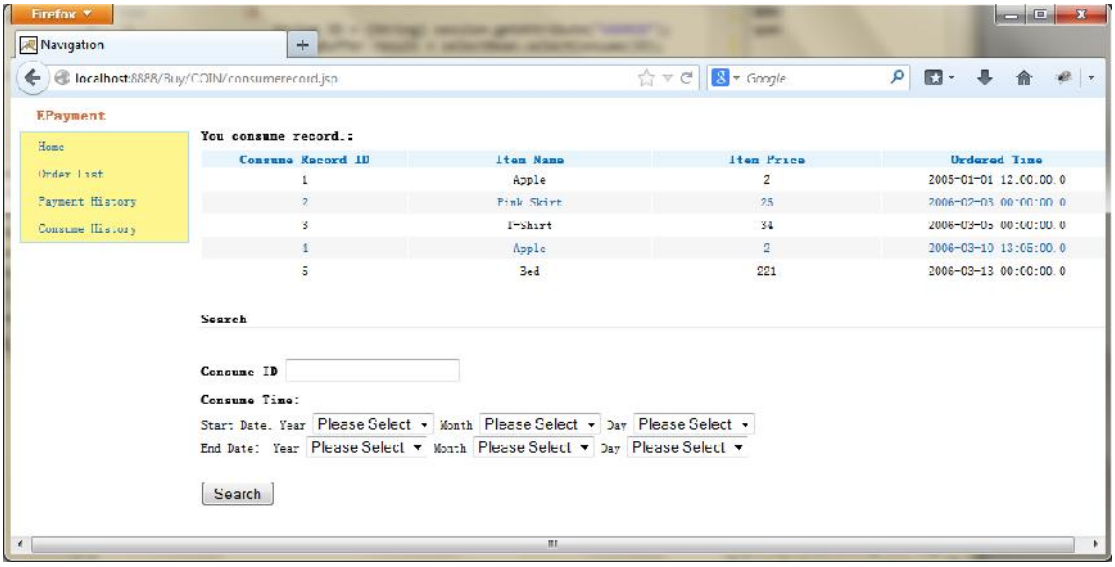
Search by Payment Types:



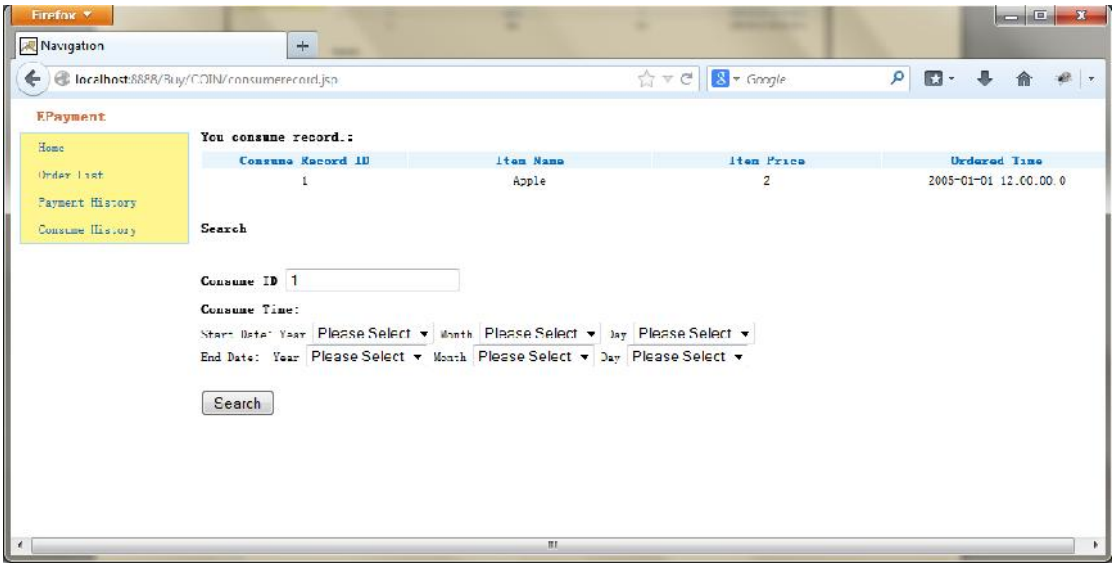
Search by Payment Time:



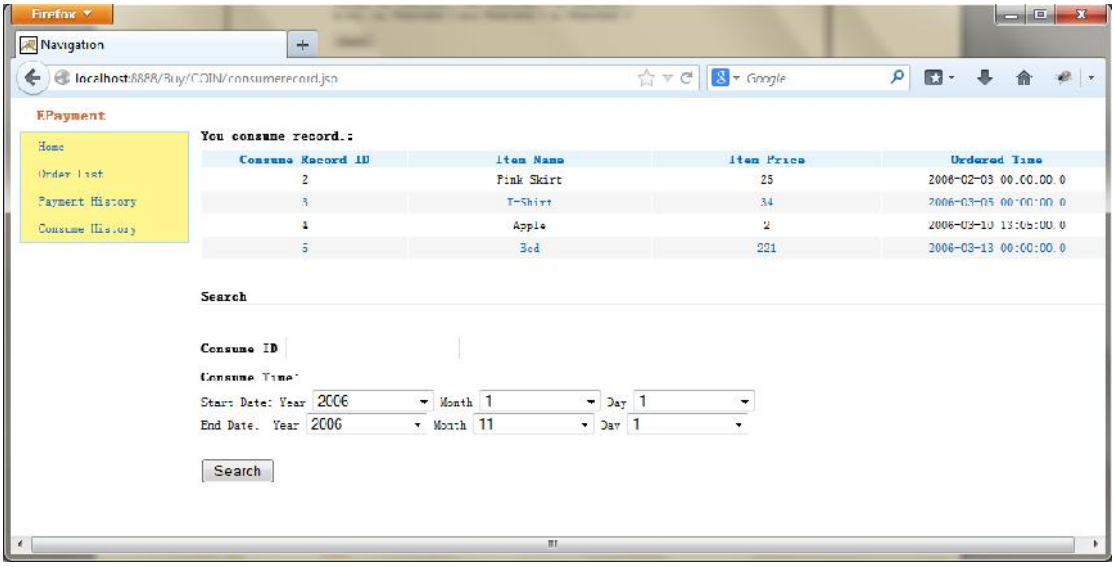
(4) Consume History:



Search by Consume ID:



Search by Consume Time:



(5) Error - Not Found: