Online Payment System

The Online Payment System is a third-party secured transactions system. Any buyer can submit payment into an account first, and then the system will notify the seller to consign the goods. Once the buyer has received and confirmed receiving their goods, the system will transfer the money to the seller's account, thus completing their web transaction.

Basic functional and non-functional requirements of the payment system are described in the rest of this document. Please make sure to read them carefully.

i. Functional Requirements

✧ **Module 1: Personal Account Management** Allow a basic user (i.e. buyer or seller) to register an account with his/her basic personal information such as real name, ID number and email address, and modify this information at any time. Users can be also allowed to process basic capital account operations, including setting and changing payment passwords, charging accounts, and querying account balance. Users can check their payment-records (statements) which will be organized and displayed in two forms: monthly and yearly. Users should be able to utilize all of these functions through a main interface, and links to other function modules should be available as well.

✧ **Module 2: Payment Transaction Processing** Allow users to check current order information, including total number of orders, list of items ordered, transaction amount, buyer and seller information, and order status. Buyers should be able to process such operations as payment and refunds, while sellers can confirm that the goods have been shipped. Trading history queries and complaint functions should be also implemented. Furthermore, the transaction flow should be recorded, as it will be used for account reconciliation and auditing. The interface of this module is needed. In the page, transaction record should be classified and displayed according to the transaction status (e.g. Processing, Non-payment, Waiting to ship, Waiting to confirm, Refunded, completed and failed) and time (e.g. Today, Last week, Last month, Last three month, Last year, one year ago). Transaction security authentication will be regarded as a plus.

✧ **Module 3: Online Booking** Hotel and flight online booking is a value-added service of the online payment system. Users are able to use this service to look up hotel and flight information, book rooms and tickets as needed, and use the their accounts to make payments. The basic operations include hotel and flight searches, information display, booking and payment, comment and scoring, and booking history query. The information of discount flight tickets and special offer rooms will be listed on the right side, when users make hotel and flight searches.

The hotel search result can be sorted by price level, hotel star level, hot level and customer-feedback score, the flight search result can be sorted by price, flight time, none-stop and Airline Company. Note that a new role, Booking Service Administrator, should be added to the system. His job will be to manage and maintain hotel and flight information through a backstage system interface, which should be offered with the main online booking module interface.

- ✧ **Module 4: Account Reconciliation and Audit** Allow auditors to run account reconciliation and audits with the transaction flow recorded in the database when transactions are completed. The system will generate the reconciliation data list in a proper form of last day (0:00 – 24:00) in a fixed time (e.g. 1:00am) every day. The list should include the order number, buyer and seller IDs, total amount, order status, and trade time of each transaction. In the event of an error or something to need double check, it should be logged at once, and auditors should be warned. The interface of this module should be well designed.

- ✧ **Module 5: System Administration** Allow administrators to maintain the system and ensure it runs smoothly. Administrators can perform the following duties through the backstage interface: add new administrators (including system and online booking), maintain administrator information, manage users (regular user, VIP user, and auditor), verify real names, and manage arbitration and blacklists. In detail, authority management should be well designed and implemented. Once someone's authority has been changed, corresponding effect should be reflected immediately. Besides, developers should build and maintain an ID database used for automating real name verification.

- ✧ **Other functional requirements for whole project:**

A. VIP mechanism: For users, the more using experience they have, the more growth points they will gain. The points rule should be defined rational and reasonable. Regular users will grow into VIP users once their growth points reach a pre-set value. VIP users own some particular privileges such as discount on booking deposit, higher limit on ticket or room amount etc.
B. Notification: Notification system should be designed in an overall view. No matter which page the user is on, he/she will be notified in a short time as long as following events have occurred: transaction status has changed; booking has succeeded or failed; real name verification has been approved or not; arbitration result has been declared.
C. Account charging: The form of prepaid card is recommended. The charging operation will be available in module 1, and also can be required when payment happening in module 2 and 3. The prepaid card number and password validation should be implemented in module 5.

ii. Non-functional Requirements

✧ **Security**

A. Privacy: User names and passwords for authentication purposes should prevent unauthorized users from accessing the system. Access controls should be built to prevent legitimate users from using system resources illegally. Some sensitive data, such as user name, password and capital amount, should be encrypted when they are being exchanged. The password should be encrypted before it is stored. During user login, SQL injection, password forcible cracking and forged session intrusion should all be prevented.

B. Integrality: Prevent unauthorized users from modifying and inserting data accidentally or maliciously. Data loss should be prevented.

✧ **Performance**

A. System Configuration: CPU 2.6G, Memory 2.0G, Disk 7200rpm.

B. Visit Capacity: Support 500 users concurrent access at least in the same time.

C. Capability: Support 10000 transaction records in total at least.

D. Response speed: For single user access, web response time < 1s, information search response time < 2s. For 500 users concurrent access, web response time < 2s, information search response time < 5s.

✧ **Availability**

A. Multi web browser support: The system should display and perform correctly in most of popular web browsers including Firefox and IE etc.