FIT5183 Programming for Distributed, Parallel and Mobile Systems

Assignment One, Semester 2A, 2014

Weiwei SUN ID: 25467247

PART II: User Guide

To begin with, the High Distinction level is expected by my job with my hard work. Firstly, all the parts of my assignment will follow the principal of software engineering and the Java code conventions. Secondly, the project is built in a quite clear structure that followed the principal of MVC, DAO, and HOPP. Thirdly, the project has clear business logic which makes it easy to modify and expand it.

2.1 System deployment

2.1.1 Deployment environment

Operating system: Windows, Linux (installed with JDK1.6 above)

Database: MySQL community server 1.7

2.1.2 Import the database

Database username: fit5183a1 (no password)

In order to using the system, user should use the following commands to import the sql file into MySQL.

```
mysql -u fit5183a1 -p < id25467247.sql
```

2.2 System startup

User should use the following commands to run the proxy server and the three airline servers.

java –jar Proxy.jar

java –jar Airline.jar cea

java –jar Airline.jar qan

java –jar Airline.jar ac

java –jar Client.jar localhost

After the startup of all servers, the client should startup in order to send request to the servers. If the client starts up successfully, the message in CMD will be similar to what shows in Figure 9. In this project, only a console user interface is supported, just as we talked before, it is easy for someone else to develop the GUI client for this system.

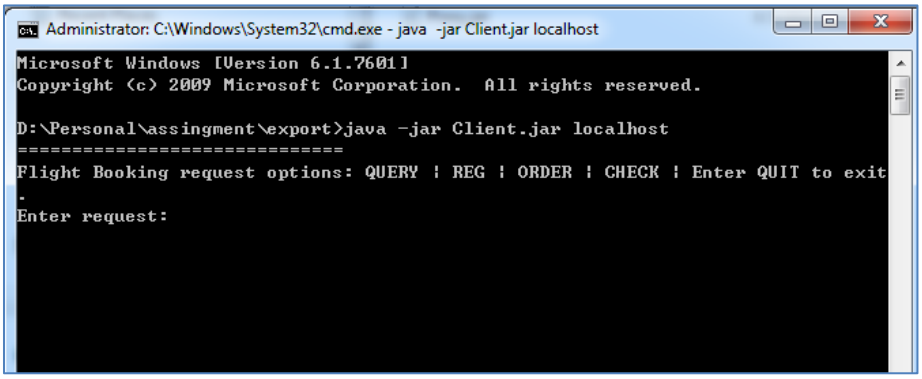

Figure 9: Client startup

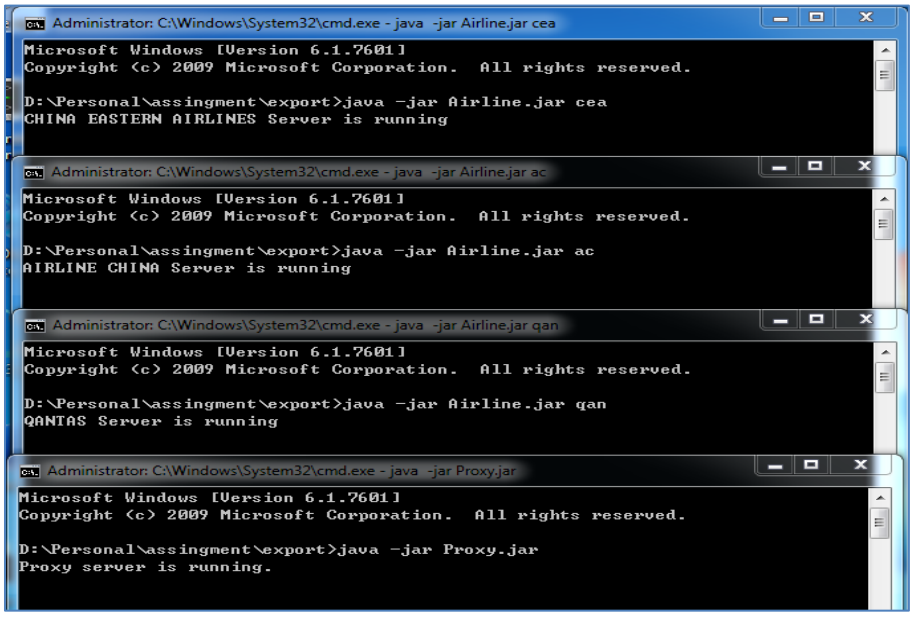

Figure 10: Server startup

2.3 System operation

2.3.1 Query flights from the servers

For the purpose of using the system effectively, user should follow some specific rules. In a flight booking system, flights querying is one of the most important functions. You can query one-stop tickets and return-trip tickets in this system.

QUERY	*fromCity*	*toCity*	*departDate*	*returnDate*

QUERY	*fromCity*	*toCity*	*departDate*

Figure 11: Query format

Figure 11 shows the format of command query, and Figure 12 shows one example of this command.

E.g. ***query*** *shanghai Melbourne 2014-03-13 2014-03-20*

```
==============================
Flight Booking request options: QUERY | REG | ORDER | CHECK | Enter QUIT to exit
.
Enter request:
query shanghai melbourne 2014-03-21 2014-03-29
Request was query shanghai melbourne 2014-03-21 2014-03-29
The available flights are:
CEA010329 [Departure]2014-03-29 [Tickets]30 [price]100
AC030121 [Departure]2014-03-21 [Tickets]30 [price]100
AC010329 [Departure]2014-03-29 [Tickets]30 [price]100
QAN010329 [Departure]2014-03-29 [Tickets]30 [price]100
==============================
Flight Booking request options: QUERY | REG | ORDER | CHECK | Enter QUIT to exit
.
Enter request:
```

Figure 12: Query round-trip tickets

2.3.2 User register

User should register his (her) personal information first before he (she) booking the specific flights. In this system, user should use the REG command to register the personal information into a specific airline database.

There is one point need to pay attention, the part of airline only support “QAN, CEA, AC”, “QAN” stands for Qantas, “CEA” stands for China Eastern Airlines, and “AC” stands for Airline China.

REG	*airline*	*name*	*phone*	*email*	*creditCard*

Figure 13: Register format

Figure 13 shows the format of command register, and Figure 14 shows one example of this command.

E.g. ***reg*** *qan john 18895730987 john@live.cn 23432438*

Figure 14: Register user information to a specific airline

2.3.3 Order tickets

User should order the specific flights after flights querying and personal information registering. In this system, user could order one-stop tickets or return-trip tickets.

ORDER	*fid1*	*fid2*	*username*

ORDER	*fid*	*username*

Figure 15: Order format

Figure 15 shows the format of command order, and Figure 16 shows one example of this command.

E.g. ***order*** *cea030110 cea010315 john*

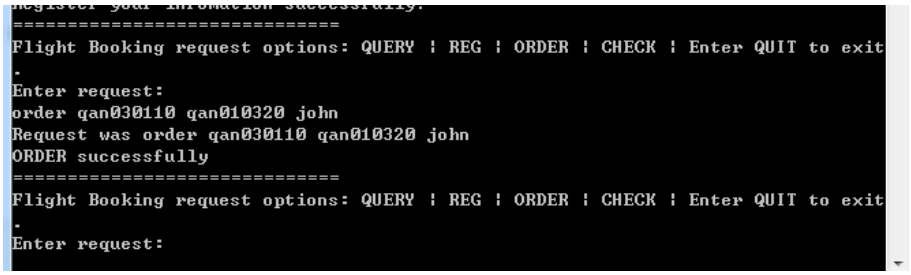

Figure 16: Order two tickets

2.3.4 Check orders

User could check his (her) orders after he ordered the flights in this flight booking system. User should check the orders by specified the airline, actually, only "CEA, AC, and QAN" are supported in this system.

CHECK	*airline*	*username*

Figure 17: Check format

Figure 17 shows the format of command query, and Figure 18 shows one example of this command.

E.g. ***check*** *cea john*

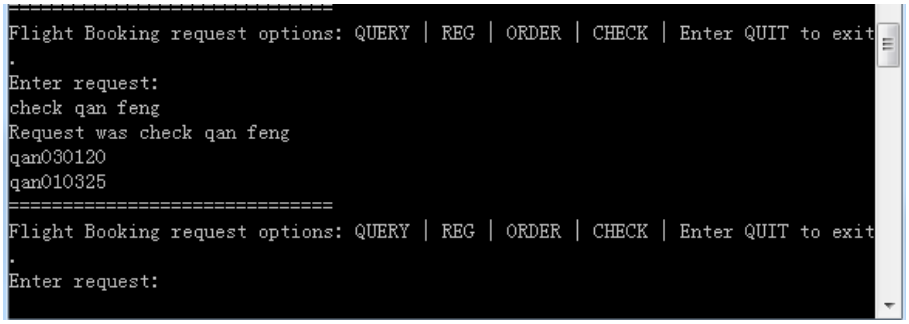

Figure 18: Check tickets

2.3.5 Quit the system

After all the operations, user can quit the system by enter the command of "quit".

E.g. ***quit***