Based on netease train ticket booking system design and implementation

YunliCheng
Guangzhou Nanyang Polytechnic College, Guangzhou 510925, China

Abstract: With the continuous development of mobile Internet technology, the use of the traditional Internet has been unable to meet the needs of people to buy train tickets online, people more hope to be able to buy train tickets anytime and anywhere through mobile phones and other terminals. Currently, the train ticket purchase software in the market does not provide the ticket snatching function of the mobile client. Based on Freemarker and Spring Model View Controller (Model View Controller), this project designed and developed its own server-side framework to improve development efficiency and enhance system security. By extending the Spring framework, the mature application of Aspect Orient Programming (Aspect orientation Programming) and annotation technology in the industry is widely used to reduce the coupling between modules, improve the development efficiency and reduce the maintenance cost. Through the extensive use of load balancing technology and the independently developed load balancing module, a stable and efficient cluster system is established in the background, which also significantly improves the user access speed. Through the independently developed monitoring platform, a large number of background servers are monitored. By widely using Squid, MenCache and other caching technologies and page static technology, the pressure of accessing the background database and file system is greatly reduced, and the speed of user data reading is significantly improved. At present, the project has been put on line. During the period when the system was put on line, the user growth was stable, the number of ticket users continued to increase, and the user experience was constantly enhanced.

1. Introduction
At present, the demand for train tickets in the market is on the rise, which is generally reflected in the continuous railway infrastructure construction, the increasing total mileage, the rapid construction of high-speed railway, the increasing number of train trips, the Spring Festival travel rush and other periods of tight demand, ticket purchase is still in short supply. The maturity of traditional Internet promotes the development of mobile Internet and provides many basic conditions for the development of mobile Internet.

Currently on the market already has a lot of plug-in software and rob ticket train tickets for tickets, but the functions are based on the form of a browser plug-in, such as 360 browser rob ticket plug-ins, in terms of mobile client, recently launched the 12306 train mobile phone client, but only has the function of normal to buy a ticket, in the aspect of mobile client haven't rob function. Therefore, it is imperative to develop a mobile app for train ticket purchase with the function of ticket snatching, so that mobile phone users can snap up train tickets anytime and anywhere with their mobile phones. The establishment of netease train ticket project, based on Freemarker and Spring MVC (Model View Controller) Model View Controller, designed and developed its own server-side framework, which improved the development efficiency and enhanced the system security. It is conducive to creating a
healthy and good railway ticket purchasing environment, establishing an efficient and convenient
ticket purchasing order, and the functions of booking and snatching tickets and monitoring the
remaining tickets let the users with demand know the information of the remaining tickets in the first
time, adjusting the balance between the supply and demand of railway tickets, and effectively
relieving the situation that one ticket is hard to get during the Spring Festival travel rush.

Nowadays, smart phones, tablet computers and other mobile devices are becoming more and more
popular in the market, which makes mobile Internet gradually become the most intimate media for
people. Through it, people can not only enjoy network information services anytime and anywhere,
but also complete many creative business activities.

2. Research contents of this paper
This paper expounds the key technologies used in the system and explains the technical route adopted
by comparison. The system requirements are analyzed from functional requirements and
non-functional requirements. The general design of the system is described, including the overall
system architecture design, functional module division and database design. The detailed design and
implementation of the system, and finally verify the system and the results show. The system structure
is shown as follows:

![System Structure Diagram]

The system is divided into two parts:
- (1) foreground function module
  A. User registration (visitors to the site need to register for a series of reservation-related activities
     in the system)
  B. Ticket enquiry (passengers check the ticket status of each train by origin, destination and date)
  C. Ticket sales (select the required ticket, fill in the purchase information, such as: id number, and
     submit)
  D. Change ticket (the user changes the ticket information of the paid order)
E. Refund (the user returns the paid order ticket information)
F. Information browsing (viewing various information published by the administrator)
G. Message board (users can post messages in the message board after logging in, and users who are not logged in can only browse messages posted by other users)
H. Personal information modification

(2) Background system management
A. User management (manage user information of the system, add, delete, change and check)
B. Ticket management (add, delete, change and check train ticket information)
D. Information release (administrator releases announcement type news and picture type news)
E. Link
F. Booking information
G. Message center management
H. Order statistics
I. Sales inquiry
J. Personal information modification

3. Netease train ticket booking system function module
The functional requirements of the system are analyzed. The system can be divided into ten functional modules: order module, push module, system timing task module, user information management module, train ticket inquiry module, red envelope distribution coupon module, partner docking module, inquiry module, ticket snatching module and user feedback module. First, the overall requirements of the system are analyzed, and then the functional modules are analyzed one by one.

Netease train ticket main components include:
(1) User information management module. This module is used to maintain the user's basic information and permission information.
(2) Train ticket inquiry module. This module can query the train number and other ticket information between two stations, so as to facilitate the user's subsequent ticket purchase operation.
(3) Partner docking module. This module realizes the mutual interface call with the partner, and completes the inquiry of booking rules, verification of remaining tickets, order ordering, cancellation of orders, application for refund of tickets, payment notice, lock order of cooperation direction system, confirmation of the existence of tickets, refund of difference, full refund, refund of tickets, purchase results and other requests.
(4) Train ticket order module. This module is the core module of train ticket purchase, controlling the transfer of money between user accounts, intermediary accounts, promotion accounts and partner accounts.

MVC, short for model-view-controller, is a common design pattern. MVC reduces the coupling between the business logic interface and the data interface and makes the view layer more dynamic. Struts is an implementation of MVC that uses servlets and JSP tags (part of the J2EE specification) as part of the implementation. Struts inherited the features of MVC and made corresponding changes and extensions according to the features of J2EE. The working principle of Struts is shown in the following figure:
Struts framework has the advantages of modularity, flexibility and reusability of components, and simplifies the development of web applications based on MVC.

Struts architecture:

Let’s look at the three parts of the component framework in the struts framework from an MVC perspective: the model, the window, and the controller. In struts framework, the model is divided into two parts:

1. The internal state of the system
2. Operations that can change state (transaction logic)

Internal state is typically represented by a set of ActinForm javaBeans. Depending on the design or application complexity, these beans can be self-contained and have a persistent state, or they can only fetch data (from a database) when needed.

Large applications typically encapsulate transaction logic (operations) within methods that can be invoked by beans that have state information. Take the shopping cart bean, which has information about the items the user has purchased, and perhaps the checkOut() method, which checks the user’s credit card and sends the order information to the warehouse.

In small programs, actions may be embedded in the Action class, which is part of the controller role in the struts framework. This works well when the logic is simple.

4. Orders module
The demand of order module is the core functional demand of the system. After ordinary users enter the application, they can complete the ticket purchase requirements in the order module. The user can query the train number, add a passenger, submit the order, pay and cancel the order.
When the user wants to buy train tickets, the prerequisite is that the user enters the ticket order function page of the application, enters the query of origin, destination and departure date, and clicks the query of train number to enter the query of train number use case, and the return result is the train number list information. The extended event stream to query the train number use case is as follows: the system shows that no qualified train number list information is found; Network connection error, please try again later; Train number list information displayed successfully.

![Image](image.png)

**Figure 3 Order sending system**

The trigger condition of the order submission use case is that the user clicks the order submission button, the pre-condition is that the user has filled in the order information, and the post-condition is that the order submission is successful and the user enters the payment page. The extended event flow includes: the order information is filled in incorrectly, the system prompts you to fill in again; Abnormal network connection, order submission failed; Order submitted successfully.

The trigger condition of the user payment use case is that the user clicks the payment button, the precondition is that the user submits the order successfully, the postcondition is that the user pays successfully and the system prompts that the ticket is being issued. Users can choose a variety of ways to pay: netease alipay pay, alipay pay, online banking payment. The extended event stream includes: abnormal network connection, payment failure, system prompt please check the network and pay again; The system is working hard to cancel the order for you. The trigger condition of the use case is that the user clicks the cancel order button.

### 5. Train ticket timing query module

Train ticket query module for the user to provide a query train number function. The participating role is the ordinary user. The user enters the query page and selects the originating station, destination station, query date and train number category through the control. After the selection is completed, click the query button to display the train number information. The use case diagram of train ticket inquiry module is shown in figure 4. Selecting the originating and destination use cases, selecting the query date use cases, and selecting the vehicle number category use cases are all done through control selection, and they have the common prerequisite of entering the vehicle number query page. The trigger condition of the use case is to click the query button, the precondition is to fill in the query information, and the postcondition is to successfully display the query list.
Figure 4 Schematic diagram of train ticket timing query module

Ticket module for the user to provide ticket grab function. The participating role is the ordinary user. After entering the ticket snatching page, the user will first check the train number, and then fill in the ticket snatching list. After filling in the ticket snatching list, the system will keep snatching tickets for the user, or click the stop button to stop snatching tickets.

6. **Database design and implementation**

The order contains the necessary data, and the data required by the order details page is ready to be packaged into MISC large field storage. The key query core data is mapped to the corresponding library table field.

Ticket and passenger information mainly exist in MISC field, label items: train number, time, departure, arrival station, ticket booking type (round trip, ordinary, seat type (hard seat; For hard sleeper; For hard lying; For hard lying down; As the soft seat; For soft sleeper; For soft sleeper; For soft sleeper; For business; For sightseeing seats; For first-class package seats; For the principal seats; First class seats; Second class; For advanced soft sleeper; Is advanced soft sleeper, ticket price, purchase quantity, adult quantity, children quantity, preferred seat, acceptable seat (allow field is empty). The storage fields are: passenger type (adult, child, passenger name, certificate type, certificate number, ticket type, birthday, number of copies of insurance, if there are several separated by '|').

Through the extensive use of load balancing technology and the combination of independently developed load balancing module, a stable and efficient cluster system is established in the background, which also significantly improves the user access speed. The functional modules of the system are clearly divided. By establishing entity relation, the whole database is designed and the redundancy of database table is reduced.

7. **Conclusion**

The 21st century is a high-speed, fast era. In this era of rapid development, speed has become our first pursuit of the goal; The second is to improve work efficiency; The third is to cut costs. Therefore, the high speed, high efficiency, high convenience and low cost of office automation have become our purpose, so I developed this train ticket ticketing system. This system can greatly simplify refund, ticket, inquiry and other very tedious work, improve work efficiency, and is easy to learn, easy to use, easy to meet customer needs.

Netease train ticket product technology platform’s load capacity can support the increasing number
of visits to the product, and its flexibility and scalability can support the complex and changing business requirements of the product, which is a safe and stable technology platform to rely on.

The technology, framework, load balancing and clustering, caching technology, page static technology and asynchronous technology used in the research and development of netease train ticket system have good security, load balancing and fault-tolerant capabilities, and can defend against strong network attacks. In the field of mobile development, netease train ticket has developed the application version of android and platform, with good platform compatibility and performance. In addition, netease train ticket client independently developed a powerful and stable script engine technology, which can control the whole access process with scripts downloaded from the server, greatly increasing the stability and availability of the client.

Figure 5 Entity relation diagram of train ticket

Author
Cheng Yunli (1980-), female, master, Communist Party member, associate professor, senior engineer, research direction: computer application technology, digital campus.

Acknowledgement
Fund Project 1 This article is the Guangdong Provincial Department of Education's Guangdong Province Characteristic Innovation Project (Humanities and Social Sciences) -- "Research on the Innovative Talents Training Model under the Background of New Engineering" (2018GWTCX054), phased research results;

Fund Project 2 This paper is the mid-stage research result of the project of “Big Data and Intelligent Computing Innovation Research Team (NY-2019CQTD-02)”, the third batch of innovative strong school project of Guangzhou Nanyang Polytechnic College in 2019.
**Fund Project 3** This paper is the phased research result of the professional construction project of Computer Application Technology (NY-2019CQZY-04), the third batch of innovative strong school engineering brand in Guangzhou Nanyang Polytechnic College.

**References**

[1] Li gang. Struts2 authoritative guide: MVC development based on WebWork core [M]. Beijing: electronics industry press, 2017.9.

[2] Cay s. holstmann, GaryComell, translated by ye naiwen et al. Java2 core technology volume I: basic knowledge [M]. Beijing: machinery industry press, 2016.5.

[3] Cay s. halstmann, GaryComell, translated by Chen haopeng et al. Java2 core technology volume II: advanced features [M]. Beijing: China machine press, 2016.3. Xia xin, cao xiaogang, tang yong. Hibernate[M]. Beijing: electronic industry press, 2017.6.

[5] zhang xiaoxiang. Java employment training course [M]. Beijing: tsinghua university press, 2018.5.

[6] sa shixuan, wang shan. Introduction to database system [M]. Higher education press, 2019.2. Li gang. Lightweight Java EE enterprise application practice: Struts2+Spring+Hibernate integration and development [M]. Beijing: electronics industry press, 2018.11.

[8] Allen. Workflow: An Introduction[M]. WFMC External Relation Commitee. 15-38.

[9] Eric a. mith Active server pages[M]. Beijing: electronic industry press, 2017.

Wang Chen. Research on structural data mining and processing problems [J]. Communications of Chinese computer society. 2019,3(3)

[11] Jarke M Koch j. Query optimization in database systems[J]. Computing Surveys. 2018, 16