Design of university fixed assets file management system and its security mechanism

Yang Yang
Shandong Womens University
Changqing, Jinan
Shandong, 250300
China
yysdwu@yeah.net

Abstract. In the 21st century, the rapid development of economy in China promotes the development of higher education. The expansion of university enrollment and school scale increases the number and type of fixed assets in colleges and universities, which generates large pressure on fixed assets management in colleges and universities. Therefore, a stable and reliable fixed assets files informatization management system is needed to help universities and colleges to solve the management problems.

A university fixed assets file management system was designed from the perspective of informatization management of fixed assets. Firstly the functional structure of the system was designed, and the technical structure of the system was constructed using JavaEE tool and MVC structure. Then system logic model was established using Unified Modeling Language (UML) tool, and relevant database was set up to support the system. Security mechanism was introduced in the aspects of concept, physical security, network security and database security.

Finally the university fixed assets file management system was developed, and its functions and performance were tested. The results demonstrated that the system had stable performance and normal log-in and functions. Hence it was a relatively perfect fixed assets file management system, which can solve a series problems of university fixed assets file management and provide assistance for university fixed assets file management.

Keywords: university, fixed assets, file management, security mechanism, JavaEE.

1. Introduction

Fixed asset is an important factor reflecting the economic development of enterprises. Fixed assets file management is based on the collection, modification and updating of fixed assets file information. Scholars in China and abroad have studied the subject. Peng [1] introduced Internet of Things into assets management, described the advantages of radio frequency identification devices (RFID) compared to bar code tag, and analyzed the changes of university fixed assets file management after the application of Internet of Things as well as the limitations of Internet of Things in assets management. The research results demonstrated that Internet of Things was useful in enhancing the efficiency of university fixed assets management. You [2] investigated the application of RFID technology in

university fixed assets management and considered RFID could accelerate fixed assets accounting and strength the dynamic management of fixed assets. Wu et al. [3] put forward an open-type architecture for fixed assets management.

The communication between different business modules and seamless integration of heterogeneous system were realized through enterprise service bus. Safe communication channel was established to ensure the reliable transmission of sensitive data on the Internet. Fragments were introduced to ensure fluent exchange of large documents and reduce bandwidth demand. The system could manage all fixed assets including intangible assets. Zhang et al. [4] applied MVC design based Struts framework into the development of fixed assets management system, studied how to realize MVC structure using Struts framework, and verified the effective coordination, favorable maintainability and extendibility of the system framework. Shvets Natalja V et al. [5] considered the problems of fixed assets analysis and the essential conditions of management in practice and found the physical wear and utilization degree of fixed assets could affect the estimation of management cost. In this study, a safe and reliable fixed assets file management system was developed using JavaEE tool to enhance the efficiency of university fixed assets management.

2. Problems existing in university fixed assets file management

With the rapid development of Chinese universities, the management and use of fixed assets are more and more standardized, but inevitably, there are some problems in the process of management. The first problem is the loss of fixed assets. Fixed assets of university include teaching equipment, books, livelihood facilities, etc.; the amount is large, and fixed asset structure is lack of [6], which brings great difficulties to management. Moreover such a large amount of fixed assets generate large pressure on account keeping. For example, subsidiary accounts may be inconsistent with general ledger, which can cause the loss of fixed asset files.

The second problem is the incompleteness of part of fixed assets files. The collection of fixed asset files is complex and some files may lose because of the infrequent communication and poor coordination between departments such as secondary schools, accounting department and logistics department and different standards for file management. The last problem is the slow updating of fixed assets files. The updating of fixed assets files is extremely difficult because of the heavy workload of filing, unified filing standard and large span of service time of fixed assets. The authors of this study considered that developing a fixed assets informatization management system could effectively enhance the management efficiency of fixed assets in universities, standardize file management and bring the values of fixed assets into full play.

3. Design of fixed assets file management system

3.1 Design of system architecture

(1) Functional framework. As shown in Figure 1, the functional structure of the system was divided into four sections, i.e. fixed assets file setting, fixed assets management, assets information query and user management. Fixed assets file setting included recording of the basic information of fixed assets including the category, producer and amount [7], the affiliated department, storage place, person in charge and assets disposal way including allocation, sale, breakage and scrap. Fixed assets management includes addition, elimination, borrowing, returning and depreciation of fixed assets [8] and recording of alteration of fixed assets related information including users, values, storage place and amount. Fixed assets information query included query of the basic information, borrowing and repairing conditions of fixed assets and other information [9]. Most of the users of the system were from the administrative departments or secondary schools of universities. System administrator can add, modify, delete and query user information to enhance the security of the fixed assets file management system.

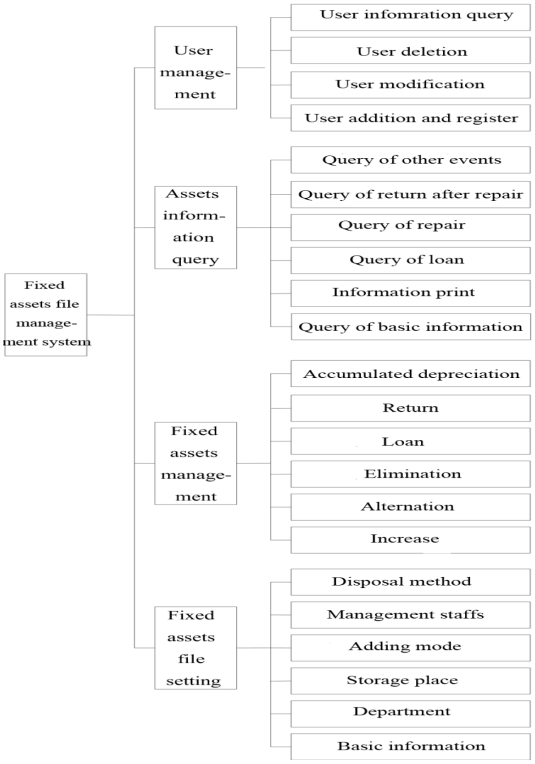


Figure 1: System function design

(2) Technical framework. The system was developed using JavaEE and java server. MVC structure [10], i.e. model-vision-controller, was used, which was transformed to presentation layer, business logic layer and data access layer. The presentation layer was responsible for designing interactive interface with Cascading Style Sheets (CSS), Extensible Markup Language (XML) and JavaApplat to display users the functions and interaction effect of the system. The business logic layer was used to do logic identification on function commands, analyze the categories of data which need to be processed, deliver instructions such as addition, modification and query of data to database. The data access layer was responsible for connecting database using Mybatis framework technology, call Structured Query Language (SQL) sentence in the database using Mapper port, conduct operations such as addition and modification according to data command, and feedback via the business logic layer and representation layer. The technical framework is shown in Figure 2.

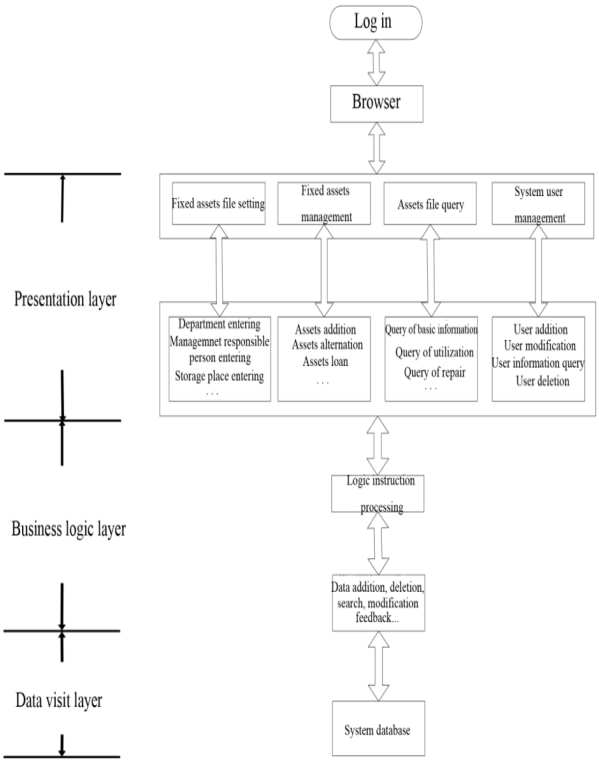


Figure 2: The technical framework of the system

3.2 Model building

Unified Modeling Language (UML) [11] was used to build model. Hierarchical design was adopted. Firstly a large class, for example, assets file, was designed,

followed by the static modeling of small class. The design of large class established the overall framework, while the design of small class added many content and details into the system framework. Figure 3 is one of the static models of fixed assets file management system. In Figure 3, fixed assets file, fixed assets, assets delivered for repairing, scrapped assets, system administrator, system user and fixed assets classification were all a class of the system. Corresponding detailed attributes were set for each class. There was a connection between every class. For example, assets delivered for repairing could be regarded as a subset of fixed asset class, fixed assets class could be regarded as a subset of fixed assets file class. System user class and system administrator class can transfer instructions to fixed assets class. Some rights of system users were limited [12], and administrator with the largest authority could query and revise assets file information.

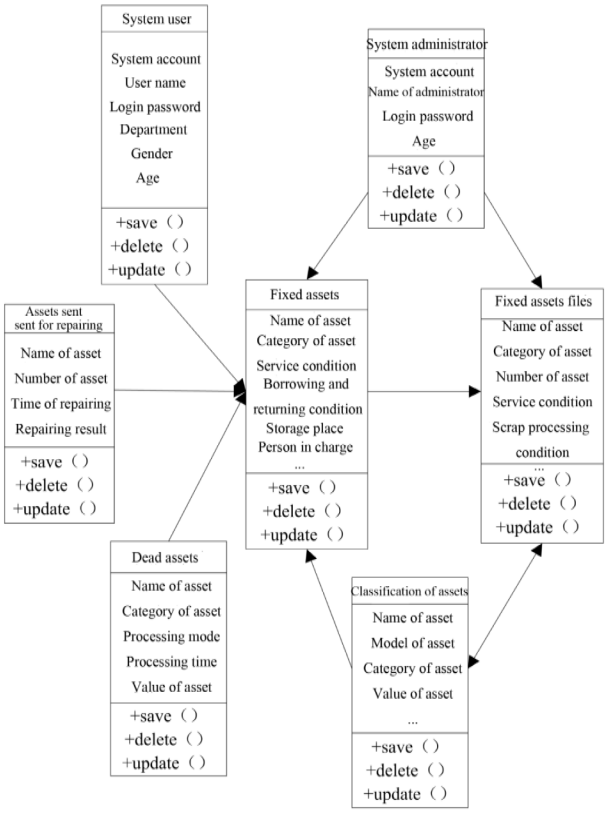


Figure 3: The static model of fixed assets file management system

A complete system generally includes many dynamic behaviors. Here those dynamic behaviors were represented by a dynamic diagram. Figure 4 suggests the dynamic model for the borrowing and returning of fixed assets. The flow

started from log-in. Then instructions were released. Next a command control node encountered. The node had two branches, one was query of borrowing and returning conditions of fixed assets and the other was query of ranking and idle amount of fixed assets. The grades of the two commands were the same. One command could be exerted only when the other has stopped. For example, a user needed to apply to the system for borrowing fixed assets. Then he was allowed to query the condition of fixed assets. Only fixed assets that meet the condition in the database could be borrowed. After all instructions finished, the user exited the system.

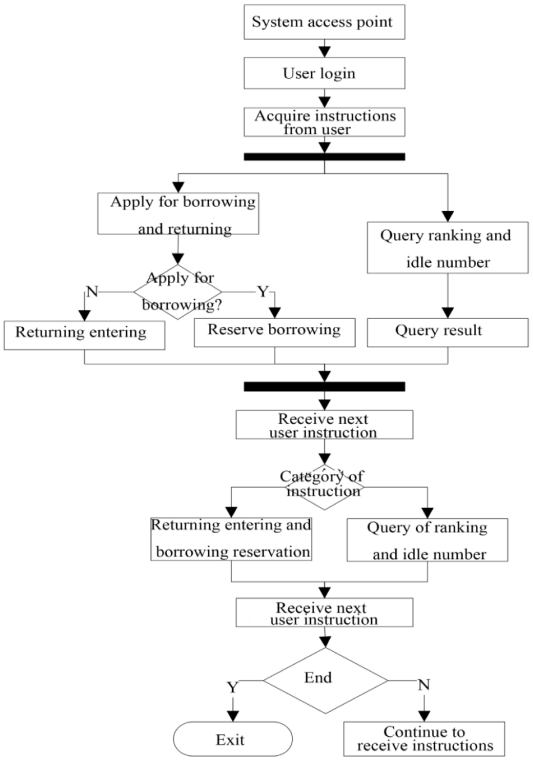


Figure 4: The dynamic model for the borrowing and returning of fixed assets

3.3 Design of database

Design of database is quite important for a system. SQL [13] was used for construction of database. Database was composed of several tables [14]. Tables were designed according to functional requirements such as user administration, fixed assets file management and analytic statistics. Some database tables were displayed. Table 1 and 2 are user table and fixed assets table respectively.

Character field	Data type	Description	Field length
Login	Name varchar	Login name	10
User	Name varchar	User name	10
Login	Password varchar	Login password	20
Department ID	INT	Department	8

Table 1: User table

Character field	Data type	Description	Field length
Declarant	Varchar	Staff responsible for declaring assets	10
Auditor	Varchar	Staff responsible for auditing assets	10
Assets name	Varchar	Name of assets	20
Assets class	Varchar	Class of assets (educationequipment, living equipment, etc.)	12
Assets model	Varchar	Model of assets	8
Assets price	Float	Purchase price of assets	8
Manufacture factory	Varchar	Manufacture factory of assets	8
Date of registration	Datetime	Date of assets entering university	10
Assets utilization department	Varchar	Department which utilizes assets	10
Spare 1	Varchar	Spare content to be added	50
Spare 2	Varchar	Spare content to be added	50

Table 2: Fixed assets table

3.4 Design of security mechanism

Security mechanism was designed in the aspects of system concept, physical security, network security and database security.

System concept: The management concept of assets file managers should be strengthened. Moreover, managers are required to be more interactive and coordinative, put an example to solve problems, establish the awareness of security and secrecy, and trained to be more careful and responsible.

Physical security: Stored files and transmission equipments should be guaranteed not to be easily destroyed by natural disasters. Moreover fixed assets files should be stored using multiple storage tools such as light disk, magnetic disk, cloud disk and electronic equipment. The security of information carrier and normal operation of system should be ensured. Generally light disk is used for long-term storage, while other storage tools are used for temporary storage.

Network security: Network security threats [15] mainly come from network attack and system bugs. Network security is protected using firewall and antivirus program.

Database security: Database access right is set as database level, record level and field level. Access right and content of users are controlled. System privileges are divided into four levels. The first level is system administrator who has the highest authority. The second level is file manager who is allowed to check and download files but needs to be authorized by system administrator. The third level is user manager who can add, delete and revise user but also needs to be authorized by system administrator. The fourth level is general user who can check field-level data of some fixed assets and can access to files of higher levels after the authorization of system administrator.

4. System implementation and test

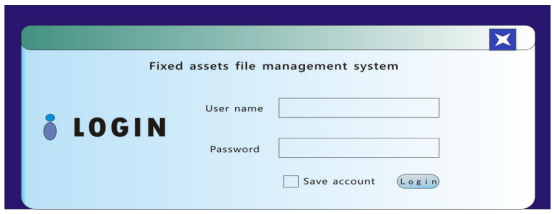
4.1 System implementation

The implementation environment of the designed university fixed assets file management system was follows.

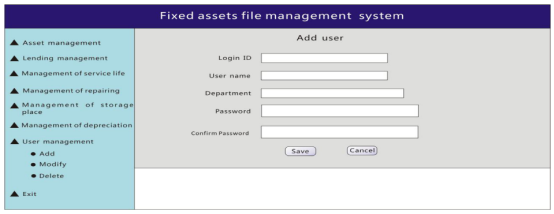
As to the server, HP DL180Gen9 which was equipped with InteXreon 1.9GHz (CPU), 8 GB internal storage, 1 TB hard drive capacity, 192.168.70.253 (local area network), Microsoft windows 2003 Server, Microsoft SQL Server 2012, 360 Security Guard antivirus program and CISCO ASA5545-K9 (FireWall Gateway) was used.

As to the client, Hewlett-Packard personal computer which was equipped with Hard drive capacity larger than 80 GB, Intel dual-core processor, internal storage larger than 1 GB and local area network was used.

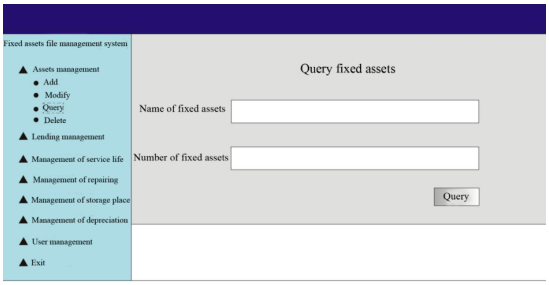
The obtained system interface is shown in Figure 5.



a. Login interface



b. User addition interface



c. Assets query interface

Figure 5: The interface of the fixed assets file management system

4.2 System test and results

Possible bugs in the system were managed and controlled using mantis v1.2.17. The stability of the system was tested using ORTHOS software. Valgrindv5.0 was used to test whether there was memory leak. The load capacity of the system was tested using LoadRunner 8.1. The utilization condition of system functions was tested using modules. Table 3 demonstrates the results of system test. As there are many module testing examples, part of test examples and test results were displayed (Table 3). The results suggested that the system operated normally and stably, users could log in the system successfully and use function as expected, no bugs were found, and there was also no storage leakage. The requirements on functions and performance were all satisfied. The stable, reliable and perfect fixed assets file management system can help universities to sort out the information of fixed asset files to make the information more completed, solve complicated archiving issues, timely update the conditions of fixed assets, and improve the management efficiency of fixed assets.

No. of test examples	Operation	Input data	Expectation	Test results
1	Input correct name of account	Account name: admin Password: admin	Login successfully and enter main interface of system	Login normally
2	Not input name of account	Account name: Password:1111	Hint name of user and password cannot be empty	Meet expectation
3	Input correct name of account and wrong password	Account name: admin Password: 1111	Hint name of user and password are wrong	Meet expectation
4	Enter user addition interface, input information, and click Save button	ID: 110 Name of user: xx Department: office of academic affairs Password: 123456 Confirm password: 123456	User addition succeeds	Meet expectation
5	Input correct asset information and click Save button	Asset information	Save successfully	Asset information is successfully saved.
6	Input existing number number of asset and click Query button	Asset number	Hint corresponding asset information	Meet expectation
7	Input inexistent number of asset and click Query button	Asset number	The queried information is not found	Meet expectation
8	Revise existing asse nformation and click Confirm button		Hint successful information edit	Meet expectation

Table 3: Some test examples and test results

5. Conclusion

In this study, the problems existing in fixed asset management in universities were summarized, and then a management system was designed against the problems. Firstly the functions of the system were designed, and the technical architecture of the system was established using Java EE and MVC model. Secondly the logistical model and database were established to support the system framework. Then security mechanisms which involved system concept, physical security, network security and database security were introduced to enhance the security of the system. Finally the system was implemented and tested using ORTHOS software. The testing suggested that the system operated stably, had no security bugs, and had perfect functions. The file management system basically satisfies the working requirements and can coordinate with managers in universities to comprehensively and timely file information of fixed assets and timely grasp the changes of fixed assets files to effectively manage fixed assets in universities. For employees and society, the research results can help file management staffs in universities process fixed assets more efficiently and make the information management in universities more orderly, which can ensure the stable development and construction of universities and promote the development of higher education.

References

[1] Y. Peng, *Prospect analysis of IOT application in fixed assets management of colleges and universities,* Advanced Materials Research, 2014, 945-949, 3581-3584.

[2] Y. Zhang, *Application of network information management in fixed assets in colleges and universities,* Journal of Harbin University, 2012.

[3] C. Wu, Z.W. Li, *Design on SOA framework for fixed assets management system*, Computer Engineering & Design, 33 (2012), 2491-2495.

[4] D. Zhang, W.X. Zhang, *Design and implementation of fixed assets management system based on the struts framework*, International Conference on Internet Technology and Applications, IEEE, 2011, 1-4.

[5] Natalja V. Shvets, Aleksandr L. Brodskiy, *Problem questions of analysis of the fixed assets of enterprise*, Business Inform, 8 (2012), 159-162.

[6] Y. Peng, *Establishing universitiesa fixed assets network information management platform based on RFID technology*, Applied Mechanics & Materials, 2013, 321-324, 2927-2930.

[7] M. Mogylova, *Institutional provision of agricultural fixed assets revaluation Up-To-Date*, Accounting & Finance, 2014, 167-172.

[8] A. Okungbowa, *Valuation of fixed assets. Asset accounting configuration in SAP ERP*, Apress, 2016, 302-316.

[9] B. Zanko, *Current conditions and development of cost accounting for repairs and improvement of fixed assets*, Accounting & Finance, 4 (2014), 24-28.

[10] H.J. La, S.D. Kim, *Balanced MVC architecture for high efficiency mobile applications*, KSII Transactions on Internet & Information Systems, 6 (2012), 1421-1444.

[11] A. Knapp, H. Storrle, *Unified modeling language*, Informatik-Spektrum, 21 (2016), 89-90.

[12] J. Tang, C.S. Xiao, J.P. Wu, *Permission centralized control system based on LDAP and module.* Applied Mechanics & Materials, 2014, 494-495, 1262-1265.

[13] H. Piyayodilokchai, P. Panjaburee, P. Laosinchai et al., *A 5E learning cycle approach based, multimedia-supplemented instructional unit for structured query language*, Journal of Educational Technology & Society, 16 (2013), 146-159.

[14] Y.L. Fan, *Database table scan and aggregation by exploiting internal parallelism of SSDs*, Chinese Journal of Computers, 35 (2012), 2327.

[15] N. Mishra, S. Sharma, R. Chasta, *Trust, security and governance in cloud computing*, Journal of the American College of Surgeons, 190 (2013), 717-723.

Accepted: 7.05.2019