3-계층 기반의 온라인 시험 체계 구현

Achievement of A Three-Tier Based Online Examination System

유추일*, 손영호*

Qiu-Yi Liu*, Young-Ho Sohn*

요 약

현재 수많은 서비스가 인터넷을 통해서 제공되며 이러한 추세는 계속해서 심화될 것으로 보인다. 전통적인 시험 방식 또한 인터넷을 이용한 시험으로 대체되고 있으며, 미국 ETS의 IBT TOEFL이 그 대표적인 예로 볼 수 있다. 뿐만 아니라 IT와 관련한 많은 시험들이 인터넷 기반으로 진행되고 있으며, 이와 관련한 논의가 활발히 진행되고 있다. 인터넷을 이용한 시험은 수험생들의 접근성을 높일 수 있으며 경제적이기 때문에 기존 출석 방식의 시험에 비해서 많은 장점을 가질 수 있다. 본 논문은 인터넷 기반의 자동 시험 플랫폼을 위한 브라우저 및 서버 프로그램을 개발하는 것을 목표로 하고 있으며, 기존의 2-계층 시스템들과는 달리 3-계층 기반으로 구현하였다. Visual Studio.Net 2005와 SQL Server 2000 개발 도구를 사용하였으며, ASP.NET 2.0 개발 플랫폼을 기반으로 하였다. ADO.NET을 사용하여 데이터베이스에 접근하였으며, 프로그램 언어로는 C#을 사용하여 구현하였다.

Abstract

Recently, various services through the Internet are gradually increased and developed. The traditional way of examination has been replacing by the online test as well. The most representative is the IBT TOEFL by the ETS in the US. Besides this, a lot of online tests and some related discussions are very fierce to carry out due to the continuous increase of the number of exam candidates. Taking account of the economic issues compared to the previous test, this online method has a lot of strengths. This paper aims to build an online test system based on the 3- tier browser-server architecture, which is different from the commonly used 2-tier based system. This system was achieved using the Visual Studio.Net 2005 and SQL Server 2000 as development tools, and based on the ASP.NET 2.0 platform, using the ADO.NET and C# language.

Key words : 3-tier, online exam, .Net, C# language

Ⅰ. Introduction

This online exam system would base on the Browser/Server architecture. Browser/Server is an important adaptation of modern network-centric computing, that optimizes the performance of mission critical enterprise applications deployed via the corporate entrance and/or World Wide Web[1].

* 영남대학교 컴퓨터공학과 (Dept. of Computer Engineering, Yeungnam University)

· 교신저자(Corresponding Author) : 손영호

· 투고일자 : 2009년 1월 10일

· 심사(수정)일자 : 2009년 1월 14일 (수정일자 : 2009년 1월 30일)

· 게재일자 : 2009년 2월 28일

The 3-tier architecture is well defined and become commonly used. Today, several applications use this architecture. When .Net arrived several developers were scratching their heads. It was a whole new platform for most of them, and a difficult and confused time seemed to begin[2]. The web Service was introduced somewhere around this time, that was a bright light for some solution architects and developers. Nowadays, we understand how to apply the 3-tier architecture on the .Net platform[2].

Currently we replace COM/COM+ components to Web Services. Thus, instead of using a binary standard for the communication, XML and HTTP is used.

Ⅱ. Related work

2-1 Present system

Most of the present online exam systems are based on the 2-tier architecture (Fig. 1). In this implementation, the three components of an application (presentation, processing, and data) are divided among two software entities (tiers): client application code and database server[2]. A robust client application development language, and a versatile mechanism for transmitting client requests to the server are essential for a 2-tier implementation[3].

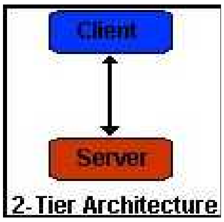


그림 1. 2-계층 구성

Fig. 1. 2-tier architecture

Presentation is handled exclusively by the client, processing is split between client and server, and data is stored on and accessed via the server. The PC client assumes the bulk of responsibility for application (functionality) logic, with respect to the processing component.

In the data access topology, the data engine would process requests sending from clients. Currently, these requests are most typically form of SQL. Sending SQL from client to server requires a tight linkage between the two layers. To send the SQL, the client must know the syntax of the server, or have this translated via an API (Application Program Interface). It must also know the location of the server, how the data is organized, and how the data is named. The request may take advantage of logic stored, and processed on the server which would centralize global tasks, such as validation, data integrity, and security. Data returned to the client can be manipulated at the client level for further sub selection, business modeling, "what if" analysis, reporting, etc[2],[3].

2-2 3-tier architecture

And the online exam system in this paper is based on the 3-tier architecture.

3-tier architecture(Fig. 2) splits the processing load as shown below:

- The client tier contains the presentation logic, including simple control and user input validation.
- The middle tier is also known as the application server, which provides the business processes logic and the data access.
- The server tier provides the business data[3].

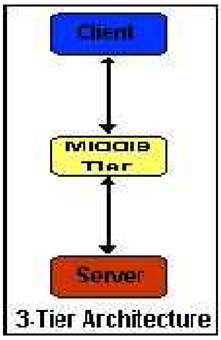


그림 2. 3-계층 구성

Fig. 2. 3-tier architecture

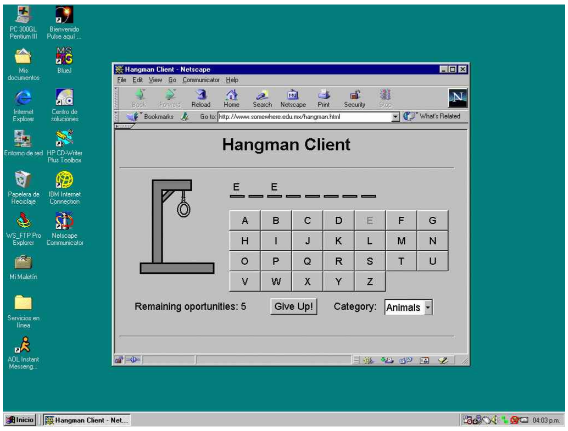
2-3 Comparison 2-tier with 3-tier

The most compelling advantage of a 2-tier environment is the speed of the application development. In most cases a 2-tier system can be developed in a small fraction of the time, it would take to code a comparable but less flexible legacy system. 2-tier architectures work well in relatively homogeneous environments with fairly static business rules. This architecture is less suited for dispersed, heterogeneous environments with rapidly changing rules.

As the bulk of application logic exists on the PC client, the 2-tier architecture faces a number of potential version control, and application re-distribution problems. A change in business rules, would require a change to the client logic in each application in a corporation's portfolio, which is affected by the change. System security in the 2-tier environment can be complicated, since a user may require a separate password for each SQL server accessed. The proliferation of end-user query tools can also compromise database server security[3-6].

The 3-tier architecture attempts to overcome some of the limitations of the 2-tier scheme by separating presentation, processing, and data into separate, distinct software entities (tiers)[6]. The same types of tools can be used for presentation as were used in a 2-tier environment; however these tools are now dedicated to handling just the presentation. When calculations or data access is required by the presentation client, a call is made to a middle-tier functionality server. This tier can perform calculations or can make requests as a client to additional servers. The middle-tier servers are typically coded in a highly-portable, non-proprietary language such as C. Middle-tier functionality servers may be multi-threaded and can be accessed by multiple clients, even those from separate applications[6-8].

In order to demonstrate these design concepts, the general outline of a simple 3-tier "Hangman" game will be presented[9]. The purpose of this game, just in case the reader isn't familiar with it, is to try to guess a mystery word, one letter at a time, before making a certain number of mistakes[9].



Hangman Client - Netscape

Go to: http://www.somewhere.edu.mx/hangman.html

Hangman Client

E _ E _ _ _ _ _ _

Remaining oportunities: 5

Category: Animals

그림 3. Windows 98시스템에서"Hangman Client"는 운행하고 있다

Fig. 3. Hangman Client Running in Windows 98

The data server is a Linux box running the MiniSQL database management system. The database is used to store the mystery words. At the beginning of each game, one of these words is randomly selected[9].

At the client side, a Java applet contained in a web page (originally obtained from a web server) is responsible for the application's graphical user interface (Fig. 3). The client platform may be any computer with a web browser that supports applet. The game's logic is not controlled by the applet; that's the middle tier's job. The client only takes care of the presentation logic: getting the user's input, performing some simple checking and drawing the resulting output[9].

The server in the middle-tier is a Java application, also running within a Linux box. The rules of the "Hangman" game (the business rules) are coded in this tier. Sockets and JDBC, respectively, are used to communicate with the client and the data server through TCP/IP[9].

The 3-tier architecture is a versatile and modular infrastructure intended to improve usability, flexibility, interoperability and scalability[9].

Ⅲ. Abstract design

3-1 System construction analysis

There are four partitions in this system (Fig. 4):

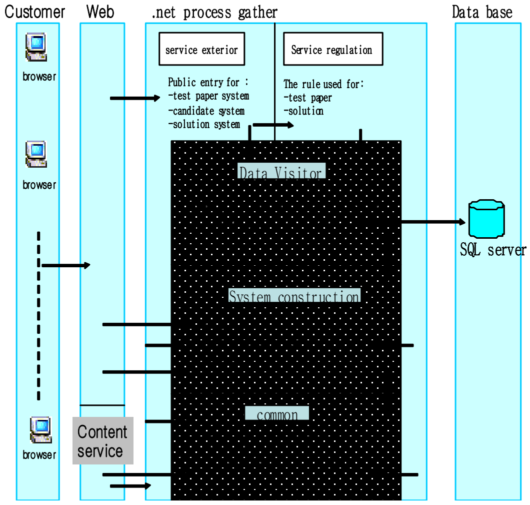


그림 4. 시스템 구조도
Fig. 4. system structural relation

- Wed level: It provides the corresponding procedure for the customer visit and opposed by the ASP.NET Web window and code hide document[10].
- Business exterior level: Provides the data for the Web level used[10].
- Business regulation level: Include some kinds of service and logic[10].
- Data accessing level: Provides the data service for the service regulation level[10].

In the Fig. 4 the “Customer” vertical and “Web” vertical are used to indicate the first tier in the 3-tier architecture. The business exterior level and business regulation level include in the “.net process gather” vertical. This vertical is the middle tier in the 3-tier architecture. At last the data access level include in the “Data base” vertical. And in the Fig. 4 the “Common” includes the data sets which for delivery the information between each layers. And the “Common” also includes the application configuration and tracking category in the “SystemFramework” item which in the “System solution program” files[11].

3-2 System functions

The main function: at first the candidate enters the system, answers the question, hands in the examination paper and then the system would grade the test paper.

Interface function: the candidate in the different status would go to the different interface.

External interface function: the test system can connect with some other system, such as bank system.

Ⅳ. Process analysis

4-1 Detailed design

According to the structure order, there are four levels in this system.

They are DA (data access) level, BR (business rule) level, BF (business facade) level, common level, and UI level (use interface).

4-1-1 DA level

The duties of this level are to complete that the entire database has accessed, include the read database and write database [12],[13]. This level can call the common level for external level, and the interface for other levels.

In this program the DA Base class has been designed at first. This class takes out the access parameter from the system installer. So the other classes in this level can use it well [12]. And then this class defines the interface class according to the persistent structure of the database.

The interface classes are IStudentData, IquestionData, IpaperData, IconfigData, IquestionCardData, and IstudentResultData.

4-1-2 BR level

The duties of this level are to complete all the BR operate, include to operate data reformatting and calculate the data. This level can call the common level for external level, and the interface for other levels [13]. Same to the DA level, the BR level need to define the interface class. And then use the interface class to achieve the interface class capsulation.

The interface classes in this level are: IpaperData, IconfigData, IexamRule, Igrade, IstudentInfo, IqueationCardData, and IstudentData.

4-1-3 BF level

The BF level is used to supply the data which can be directly used[13]. And then transmit the influent information which from the user of the UI level to other levels. The information is principal transmit to BR level to be processed. This level is used as the interface between the User Facade level and other levels.

The interface classes in this level are: IpaperData, IconfigData, IexamRule, Igrade, IstudentInfo, IquestionCardData, and IstudentData.

4-1-4 Common level

The duties of common level are to packing the fixed algorithms and fixed format describer which in the project. So every level can call these algorithms and format describer commodiously in time of need. The cohesion of these algorithms and format describe in whole project can be enhanced and degree of coupling in these also low. And then it would be extremely convenient for developer either in the developing process or later system maintenance[11-13].

This item consists of two parts. The first part is the algorithm class. There are Rc4, ValidateUtil, StringUtil, Jscript, FileIO, and FileZip. The second part is describer class which includes StudentDataSet, ConfigDataSet, PaperDataSet, QuestionCardDataSet, and QuestionDataSet.

In this paper, the algorithm classes would be explained. Rc4 class, this class is simple. This class bears the way to encryption, decrypt and data format transformation. ValidateUtil class, which to encase in the way for character string format which used in the system. In StringUtil class the operate way has been packaged which for character string. The way used to realize the special function has been packaged in the Jscript class. The operation for read and write file has been packaged in FileIO class. The way to compress the system file has been packaged in FileZip class[11],[12].

4-1-5 WebUI level

The user interface level as the operator interface and display level is used to connect with the user. We need to consider the user matter first of all[12].

The PageBase class has been designed at first.

This class is the heir to UserControl. The "session call" and "common method" has been realized as "attribute" and "method". Based on this, all the control units define by the user in the user interactive page as the heir to the basic classes of the page. And the entire elements of the page load into the control units. Then the control units which have different function and defined by the user would be borne by the page.

Ⅴ. Conclusion

The present online exam systems need to install business logic and user interface in each client. And the client accesses data on the remote server through the network. As this kind of 2-tier architecture seriously threats the safety of the system, lacks of good maintainability and scalability, and reduces the performance of the server. There is a low utilization rate of logic components in these systems and just internal information can be processed.

For the deficiencies of the 2-tier architecture, in the application system development formed the 3-tier architecture. This 3-tier architecture does not have to run on different machines. So this 3-tier architecture can resolve the software distribution question in the 2-tier architecture. And this architecture also has the advantages of easier to maintain and update, can share and reuse logical components, higher security.

References

[1] UTS, "Online Examinations," http://www.softwaresecure.com/pdf/OnlineExaminations_UTS_.pdf, 2008

[2] S. Mission, KS, "browser/server technology," http://www.browsersoft.com/browserserver/5.shtml, 2008

[3] J. Edwards, 3-tier client/server at work. New York : John Wiley, 1999

[4] B. Mains, "Introduction to 3-tier Architecture," http://dotnetslackers.com/articles/net/IntroductionTo3TierArchitecture.aspx, 2008

[5] A. O. Ramirez, "Three-tier Architecture," http://www.linuxjournal.com/ article/3508, 2008

[6] M. Chapple, "Two-tier or n-tier," http://databases.about.com/od/specificproducts/a/architectures.htm,2008

[7] M. Wang, G. Fox, M. Pierce, "Thin Client Collaboration Web Services," *AICT-ICIW'* 06, pp.148

[8] M. R. Ritschard, "Thin clients: the key to our success," *SIGUCCS'06,* Nov 2006

[9] A. O. Ramirez, "Three-Tier Architecture," http://www.linuxjournal.com/article/3508, 2008

[10] B. Mains, "Three tier Architecture with ASP.NET," http://dotnetslackers.com/articles/aspnet/Three-Tier-Architecture-With-ASP-NET.aspx, Jul.2008

[11] D. Wyatt and R. J. Oberg, Introduction to Visual Basic Using .NET. Prentice Hall PTR, 2002

[12] D. Gibson, MCITP SQL Server 2005 Database Administration Exam Guide. Osborne Media: McGraw Hill, 2007

[13] J. Mayo, "Building ASP.NET Applications with C# Builder for Microsoft.NET," http://www.informit.com/articles/article.aspx?p=169672/, 2008

Liu, Qiu-Yi (刘秋一)

2006년 8월 : 영남대학교 전자정보공학부(공학사)
2009년 2월 : 영남대학교 컴퓨터공학과(공학석사 예정)
관심분야 : 온라인 콘텐츠, 센서용 콘텐츠

손영호 (孫榮晧)

2002년 5월 : ECE Dept., Texas A&M University (공학박사)
2005년 3월~현재 : 영남대학교 전자정보공학부 조교수
관심분야 : GIS 시스템, 센서시스템/네트워크