B. ADIGÜZELLİ

M.S. Thesis

2009

ZOI: A VIRTUAL CAMPUS PLATFORM

ZEYNEP BURCU ADIGÜZELLİ

B.S.,Computer Engineering, 2005

IŞIK UNIVERSITY

2009

ZOI: A VIRTUAL CAMPUS PLATFORM

ZEYNEP BURCU ADIGÜZELLİ

B.S.,Computer Engineering, 2005

Submitted to the Graduate School of Science and Engineering

in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Engineering

IŞIK UNIVERSITY

2009

IŞIK UNIVERSITY

GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

ZOI: A VIRTUAL CAMPUS PLATFORM

ZEYNEP BURCU ADIGÜZELLİ

APPROVED BY:

Prof. Dr. Selahattin KURU (Bahçeşehir University) _____
Thesis Supervisor

Y.Doç. Dr. Taner ESKİL (Işık University) _____

Doç. Dr. Seyhun ALTUNBAY (Işık University) _____

APPROVAL DATE: 13 February 2009

ZOI: A VIRTUAL CAMPUS PLATFORM

Abstract

Virtual Organization (VO) means that two or more diverse systems can exchange and use information. It provides a flexible environment for interoperable applications. Clients are able to access other systems' information while they use their own system. There is no need to access other system. Communicated systems are essential issue for members who work at a geographically distributed. In virtual enterprise systems, resources are sharing according to the requirements and managed by system administrators. Web services and RSS feed technologies widely used in cooperate systems. They must be developed on some certain business rules. Interoperate applications are connected through Internet or Intranet.

In this thesis, a model of virtual organization has been developed. System includes three different applications. They are Administrator Panel, and two different course systems, are called Hercule Course Online and Hera Course Online. Course systems exchange information and use information. One course system is responsible for creating RSS feed format by calling a web service. This service is reached by all course online systems. It has been publish methods over Internet. RSS files include detail information about course objects. Another course system in this thesis is calling RSS file, and parse related information and then use this data in its own system.

SANAL ORGANİZASYON: VIRTUAL CAMPUS

Özet

Sanal organizayon sistemleri bir yada birden fazla sistemin birbiri arasında bilgi alışverişi yapması ve bu bilgiyi kendi sistemlerinde kullanması anlamına gelir. Bu tarz sistemler platform bağımsız iletişim kurmak için tasarlanmıştır. İstemciler diğer sistemin bilgilerine tek bir uygulamadan ulaşabilmektedir. Diğer sistemde oturum açmalarına ihtiyaç yoktur. Bu tarz sistemler coğrafi olarak farklı yerlerde çalışan kullanıcıların verimli çalışabilmeleri için tasarlanmıştır. Sanal kurumsal uygulamalarda kaynaklar ihtiyaçlar doğrultusunda paylaşılmaktadır ve sistem yöneticileri tarafından yönetilmektedir. Birlikte platform bağımsız çalışan uygulamalarda yaygın olarak Web Servis ve RSS Feed teknolojileri kullanılmaktadır. Bu uygulamalar birçok iş kuralları temel alınarak geliştirilmektedir. Platform bağımsız uygulamalar İnternet veya Intranet üzerinde birbiri ile konuşurlar.

Bu tezde, sanal course sistemi geliştirilmiştir. System üç farklı uygulamadan ulaşmaktadır. Bunlar Yönetim Paneli ve birbirinden farklı 2 tane course online sistemleridir, Hercule Course Online ve Hera Course Online. Course sistemleri birbiri arasında bilgi alışverişi yapmakta ve bu bilgiyi kendi sistemlerinde kullanmaktadır. Bir tanesi bilgileri RSS formatında hazırlamak ile sorumludur. Hazırlanan bu RSS dosyasına, diğer course sistemi internet üzerinde ulaşmaktadır.RSS dosyası derslere ait detay bilgilerine sahiptir. Diğer system bu bilgileri kendi yapısında kullanmaktadır.

Acknowledgements

I would like to thank my supervisor Prof. Selahattin KURU for consulting, suggestions, giving support and patience during project development of project and writing the thesis.

Table of Contents

Özet……iv

Tableof Contents……... vi

List of Tables ……...viii

List of Figures……ix

List of Figures……ix

List of Abbreviations…….....xi

Virtual Organization Concept ……12

1.1 What is Virtual Organization? ……...12

1.2 Evolution Model for Virtual Organization……...13

1.2.1 History of Virtual Organization……13

1.2.2 Management of Virtual Organization…….. 13

1.3 Security of Virtual Organization……15

1.4 Design and Implementation of Virtual Organization……16

1.5 What is Interoperability? ……...19

1.5.2 Interoperability of Diverse Systems……20

1.5.3 Common used technologies for Interoperability……..20

Virtual Organization Technologies ……22

2.1 Federated of Virtual Organization 22

2.2 Communication of Virtual Organizations……22

2.2 Virtual Campus Definition……...22

Technologies and Technical Background of Virtual Campus……..25

3.1 SOAP……25

3.1.2 SOAP Envelope ……26

3.1.3 SOAP Messages of Virtual Campus……..27

3.2 Web Services……28

3.2.1 WSDL (Web Services Description Language) ……29

3.3 RSS…… 31
3.3.1 RSS Structure……32
3.4 Scenario……38
Technical Development of Virtual Campus…… 39
4.1 Development Environment……...39
4.1.1 .Net Framework 3.5……...39
4.1.2 C#……..39
4.1.3 SQL Server 2005……...39
4.2 User Types of Virtual Campus……40
4.2.1 User Types……40
4.3 Software Architecture of Virtual Campus……40
4. 3.1 n-Tier Architecture……...40
Conclusion……59
References ……60
References Not Cited……..62
Appendix A CD Application and resources…..……64
Curriculum Vitae…..……..64

List of Tables

Table 3.1 Sample WSDL document in ZOI……... 21
Table 3.2 Child elements of channel tag…….22
Table 3.3 Child elements of item tag……..23
Table 3.4 Item tag elements……24
Table 3.5 Data rows of Channel data table…….25
Table 3.6 Data rows of item data table……...26

List of Figures

Figure 2.1 Communication design of Virtual Campus 16
Figure 3.1 Proxy and SOAP in XML Web Service 18
Figure 3.2 Structure of envelope 19
Figure 3.3 HTTP request 20
Figure 3.4 HTTP Response 21
Figure 3.5 WEB-based and SOAP design 23
Figure 3.6 Sample WSDL Document in ZOI 25
Figure 3.7 Sample RSS Feed 28
Figure 3.8 Flow of clients reach RSS URL 29
Figure 3.9 Providing algorithm 30
Figure 4.1 Architecture design of Virtual Campus 31
Figure 4.2 Screenshot of Admin Course define 32
Figure 4.3 Screenshot of Admin Sub Course define 33
Figure 4.4 Screenshot of Admin User define 34
Figure 4.5 Screenshot of Admin Users' Course define 35
Figure 4.6 Screenshot of Admin Students' Grades define 36
Figure 4.7 Screenshot of Admin course Materials Grades define 37
Figure 4.8 Screenshot of Admin course Assignments define 38
Figure 4.9 Screenshot of Admin course Announcement define 39
Figure 4.10 Hercule Course Online Login page 40
Figure 4.11 Hercule Course Online Assignment page 41
Figure 4.12 Hercule Course Online Announcement page 42
Figure 4.13 Hercule Course Online Materials page 43
Figure 4.14 Hercule Course Online Grades page 44
Figure 4.15 Hera Course Online Assignment page 45
Figure 4.16 Hera Course Online Announcement page 46

Figure 4.17 Hera Course Online Materials page 47
Figure 4.18 Hera Course Online Grades page 48
Figure 4.19 Relational Database Diagram 49
Figure 4.20 Structure of Course table 50
Figure 4.21 Structure of SubCourse table 51
Figure 4.22 Structure of Users Courses table 52
Figure 4.23 Structure of Users table 53
Figure 4.24 Structure of Users Grades table 54
Figure 4.25 Structure of Sub Courses Grades table 55
Figure 4.26 Structure of Users Courses table 56
Figure 4.27 Structure of Users table 57

List of Abbreviations

D&I	Design and Implementation
ICT	information and communication technology
IEEE	Institute of Electrical and Electronics Engineers-
RSS	Really Simple Syndication
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
VO	Virtual Organization

Chapter 1

Virtual Organization Concept

1.1 What is Virtual Organization?

Virtual Organization (VO) is described as two or more diverse systems that can exchange and use information. It provides a flexible environment for interoperable applications. One system is working as two systems, because it also includes other systems' information. Clients are able to access other systems' information while they use their own system. There is no need to access other system. Communicated systems are essential issue for members who work at geographically distributed systems. In virtual enterprise systems, resources are shared according to the requirements and managed by system administrators.

Some business applications used to be conducted by a single team in organizations. However it is being replaced with more than one team in time according to the business requirements. Collaboration of these teams becomes specific problem. These teams cross many institutional boundaries and administrative and they are all formed dynamically. VO defines as collaborations having these characteristics. VO is well defined in article by the editors of the "Special Issue on Virtual Organizations" of the *Journal of Computer-Mediated Communication* and is :

"A virtual organization is a collection of geographically distributed, functionally and/or culturally diverse entities that are linked by electronic forms of communication and rely on lateral, dynamic relationships for coordination. The result is a "company without walls" that acts as a "collaborative network of people" working together, regardless of location or who "owns" them. In some cases, the entities composing the organization may participate in several virtual organizations simultaneously. [and] will appear less a discrete enterprise and more an ever-varying cluster of common activities in the midst of a vast fabric of relationships."

Individual workers, teams, units which are geographically distributed are electronically linked by each other via VO system. Members can easily share data and resources in VO.

1.2 Evolution Model for Virtual Organization

1.2.1 History of Virtual Organization

Literature of the VO systems does not focus on the changing business boundaries. VO applications focus on the specific patterns. In VO systems members are dynamically change. Generally VO systems are updated due to a new members. Although ICT supports the collaboration of between these organizations, there are limitations in new organizations use of the system.

1.2.2.1 Management of Virtual Organization

Management of VO systems is a difficult part of the process because of its complexity. VO systems composed of different organizations, such as companies, network users. They all have different requirements and goals. Managers are responsible for giving a

best service to their customer in changing competitive business environment. Due to continuously changing business environment, rapid growth of Information technology and globalization of markets and trends, managers should track new changes and applicate these in their business processes. VO has emerged for offering high variety of products and services [2, 3].

VO systems have flexible structure to meet customer demands. Managers are responsible for managing goal-based activities of a VO. Management of VO environment consists of four activities [4]:

- Determine and analyze the requirements
- Tracking the possibilities to satisfy customer requirements
- Allocate tasks among members of VO
- Adjustment of activities and allocation procedure

Requirements for a system described of the systems' services. These requirements reflect the needs of the customer. VO applications are designed for combined different activities which have different means and goals. Although they have independent of each other, these activities are related to other. When analysis the system requirements, distinction of between activities should be consider. VO systems meet its specification, and the system services and behavior support the customer's requirements.

Second step of the management concerns the tracking the possibilities for satisfying requirements. Production elements describe of specific features of products and service elements. Production elements describe what a specific organization is able to produce and service elements describe what customers see and may order. The manager find the way of the translate the service elements into production elements.

1.3 Security of Virtual Organization

Protecting data and resource is important requirement for VO management systems. Every given access of system and its resources should be under control. The members of the VO should trust each other while they share information and resources. The security elements should be considered in design of the VO systems, all access rights as functions should be implemented due to multi-domain security policies. VO systems should establish a secure collaboration environment to their users. Many approaches [1, 2, 3, 8] of security management of VO system have been proposed. However some of approaches do not deal with some security conflicts. Problems of secure policies shown in below:

1. Since members of VOs dynamically changed, access control policies and related services need to be updated. Also this makes the system weaker and increase the complexity.
2. Need to be mechanism to protect privacy information of independent domains. Many security policies do not include all dynamically change domains' secure policies.
3. In VO systems have conflicts of collaboration policies. VO management systems cause potential security threat to domain policies. For instance, the user can with lover privileges may have higher policies through an identity mapping process. VO environments have large domains and it is impossible to identify and correct users' privileges.

PEACE-VO frameworks [3] have been proposed for security collaboration framework for VO. PEACE-VO framework is based on security policy. It has two concepts: implicit policy conflict and explicit policy conflict to depict the potential conditions against VO collaboration policies. Algorithm is designed for detect any conflict and policy coherence among each domain. This approach is able to protect critical domain privacy. It is implemented by two protocols; VO Management Protocol and Service

Authorization Protocol. It use the Role-Based access control (RBAC) [4] to security policies model.

Authorization and database access have important role in security policy. Administrator of VO systems should define the privileges of users into system. Administrators define the access rights of users to protect the system. For example user is allowed to access services or not, and only access defined methods. Also database roles should be defined. Users have privileges that can modify the data or just read the data.

1.3.1 Security Policies

RBAC: is a model for managing privileges. Roles have been implemented in different ways. The role concept is; establish permissions based on the functional roles in the enterprise, and then assign users to a role or set of roles. RBAC provides a powerful mechanism for reducing the complexity and conflicts within the enterprise.

Bell-La Padula

Bell-la Padula model is used for access control in government and military applications. It has a set of access control rules which use security labels on objects and clearances for subjects. Model focused on data confidentially and access to classified information. There is security rules for protection of data integrity.

1.4 Design and Implementation of Virtual Organization

The main idea behind VO implies that business processes design and implement (D&I) to meet the needs of customers. Because of that structure of business processes is need to restructuring. Analysis of the current process structure and implementation at the

needs of the business process is first step of the D&I. Information and communication technology (ICT) reduce transaction costs through planning and controlling within the organization [9]. ICT remove the boundaries among traditional work flows. Business applications increase in parallel increasing in business needs. VOs have independent organizations and this limit the design process life cycle and security. Designers analyze the main strategies of business plans to achieve to get goals in their business plans.

A shift from hierarchy towards markets is backed up by transaction cost economics that assumes that markets themselves create cost for the market transactions and that the organizational designers build hierarchical organizations to minimize the cost for transactions [7]. Searching the right partner, specifying the transactions among others cause transaction costs. Networked ICT will reduce these costs and therefore increase the range of market mechanisms.

Goals and strategies of network structure are established in the design process. VO is composed of independent business units. Each business units has different goals. Furthermore business processes can be change, new processes added in time. It is increase complexity. As a result of this, managers of VO systems should choose stable software framework against to changes. Managers should consider various issues such as security, maintainability and reusability during the design of VO system. Developers prefer to use SOA (Service Oriented Architecture) framework for designing new processes, methods and tools in complex environments. To create a successfully VO systems depend on the trust among the companies, organizations or teams. Security issues should be defining by each organization in D&I process. Trust among organizations has an important part of the VO development.

Design of the system architecture affects the performance, robustness, distributability and maintainability of a system.

1. Performance

Performance of the VO system is a critical requirement. The web services approach and SOAP protocol are used in ZOI. Execution speed of request depends on the clients' network bandwidth. Because moving resources from one system to another affect the system's performance.

2. Security

The system accessed from several different clients and thus security is a critical requirement for VO systems. The system may be accessed from several different computers. Because of that a layered structure is used to protect critical assets such as database.

3. Safety

The architecture of the VO systems should be designed that safety-related information are all allocated in a single system. This reduces the costs and problems of safety validation and provides related protection systems.

4. Availability

The architecture should be designed to include redundant components and so that it is possible to replace and update components without stopping the system.

5. Maintainability

The system architecture should be designed to perform corrects defects, meet new requirements, make maintenance easier and cope with a changed environment.

Distributed systems have the following advantages of using a distributed approach to systems development:

1. Resource sharing: distributed systems allow the sharing software resources.
2. Openness: distributed systems are generally open systems; they are designed around standard protocols that allow equipment and software from different vendors to be combined.
3. Concurrency: In a distributed system, several processes may operate at the same time on separate clients on the network. These processes may communicate with each other during their normal operation.
4. Scalability: distributed systems are scalable in that the capabilities of the system can be increased by adding new resources to cope with new demands on the system. The network linking the individual computers in the system may limit the system scalability.
5. Fault tolerance: The availability of several computers and the potential for replicating information means that distributed systems can be tolerant of some hardware and software failures. In most distributed systems, cluster system architecture is used. When failure occurs in service, the other related services work instead of down service. Then there are no completely loss services.

Distributed systems are usually developed using an object-oriented approach. These systems are made up of loosely integrated, independent parts, each of which may interact directly with users or with other parts of the system. Parts of the system may have to respond to independent events. Software objects reflect these characteristics.

1.5 What is Interoperability?

There are many complex systems in business environment. They are developed in different operating systems, each of using different programming language, and therefore have different database implemented by different interfaces. Communication of these heterogeneous systems is not easy. Interoperability has become the valid issue for sharing information among systems. Interoperability is the ability of communicate two or more systems, applications, devices and exchange information among systems.

1.5.2 Interoperability of Diverse Systems

Interoperability is the ability of diverse systems exchange and use information. According to the business requirements, interoperability has become an important software issue.

In business environment there are several diverse applications are used. However, they need to be work together in the business process.

Two different systems are used in this thesis. They have different database system and are distinct applications. They exchange and use information between each other. They have similar system architecture. Both of them developed in .Net Framework 3.5 and Xml technology used for communication of these systems. One system is responsible for creating and sends it over the internet. Another system request data and use this data in its own system.

1.5.3 Commonly used technologies for Interoperability

RSS feed and Web Service technologies are most popular technologies for interoperability. Both of them are in XML file format. As a result of that, they can be supported in many different platforms. Both technologies have been used for exchange information in this thesis.

RSS Feed

RSS feed is a technology to publish web content in an xml file format via HTTP. It can be used in several applications such as blogs, news, and audio. RSS feeds are read by

RSS readers or aggregators. When the user clicks the RSS icon in a web page, he/she is able to add this to own subscribed feeds.

Web Services

XML web service is a web-based implementation. Applications can publish its own methods or data to other applications via HTTP Internet protocol. Web services send or receive data in xml data format. Web services technology is designed to support interoperate machine-to-machine communication over a network.

Chapter 2 Virtual Organization Technologies

2.1 Federation Virtual Organization

With the rapid growth of Information technology, virtual organization (VO) applications have began to be more widely used and it is based on cooperative information system. Machine-to-machine strategy plays an important role in design and implementation of VO. The characteristics of the VO applications enable the sharing of resources and information cooperation among distributed applications. VO applications contribute to the needs of the business environments, such as use of information and communication technology, time-based competition and decentralization. VO systems provide some business opportunities to companies, teams, and departments. Companies linked to other companies systems share skills and costs.

2.2 Communication of Virtual Organizations

Electronic communication in VO enables to link distant, departments, and organizations. System manager of VO use Internet technologies to communicate systems depending on the business requirements.

2.3 Virtual Campus Definition

Virtual Campus can communicate diverse systems and organization. It is also able to work them together. Three different systems are used in the development of Virtual Campus. These systems are Admin Panel and two different course systems, which are Hercule Course Online and Hera Course Online. The main idea of this system is when student enters in one of the university; he or she can reach other university's lecture resources. Figure 2.1 shows the basic structure of Virtual Campus. Both Hercule Course Online and Hera Course Online are university course systems that are used by students

and instructors. Students can reach online resources of lectures, such as Assignments, Materials, quizzes, grades and announcements by using these systems. Hercule Course Online is responsible for providing data for other universities with specific XML standards. Hercule Course Online publishes this data over the internet. Hercule Course Online transfers data by using RSS feed technology. Hera Course Online reads this data and uses it in its own system.

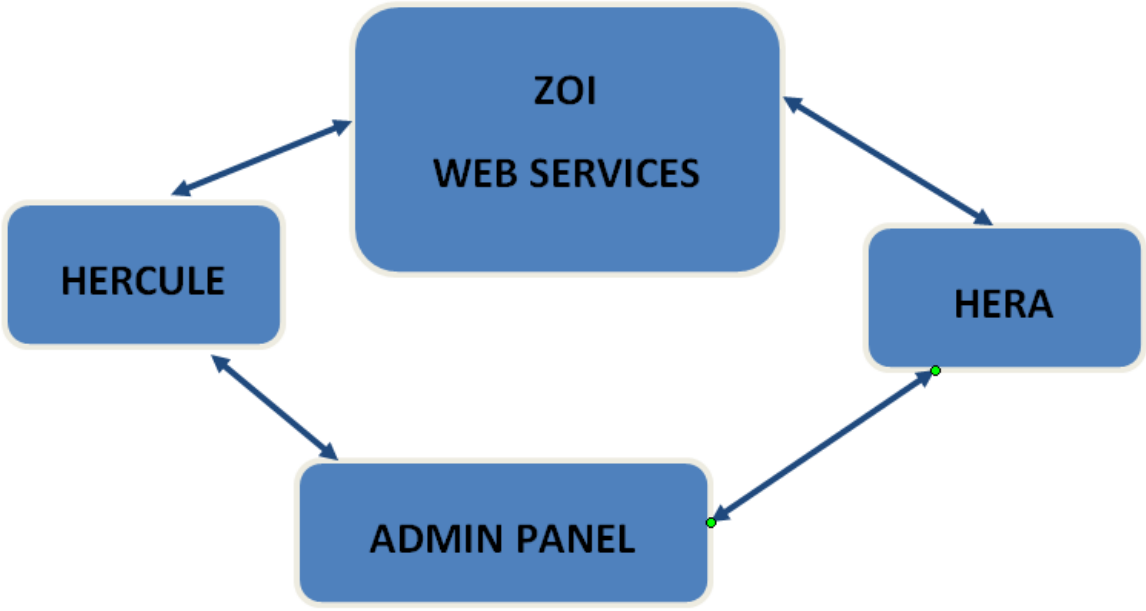


Figure 2.1 Communication design of Virtual Campus

The purpose of this system is that the student able to study in two different universities at the same. In addition, the student will be able to reach other university resources by using Virtual Campus.

2.4 ZOI Architecture

In ZOI, clients and servers are different. Clients receive services from server and not from other clients. They do not request services from other clients. Clients must know the services description and how to contact these services.

A number of clients on a network and communicate through middleware. Middleware means object request broker. It provides a set of services that allows objects to communicate.

2.4.1 Advantages of ZOI

The advantages of ZOI model are:

- ZOI has no database security problem. Clients and services do not have access rights to connect other clients' database. ZOI web service has two different methods, which are MakeRSS and GetData. MakeRSS provides a RSS with a related data. Also when the other clients need to access this data, GetData method only collect information from related RSS and then send output in XML format.
- Performance of ZOI depends on the clients' network band-with. Number of clients can access the services' methods at the same time.
- Students can access other system resources. There is no need to authenticate other system.
- There is not system language dependency. The clients system may be implemented using different programming languages, the systems may run on different platforms.

Chapter 3

Technologies and Technical Background of Virtual Campus

3.1 SOAP

Client applications call the web services methods. This request is sent to proxy object. Proxy sends this request to web service. Web service interprets this request and sends response to proxy object. Finally proxy object sends result to client applications. This flow is shown in Figure 3.1:

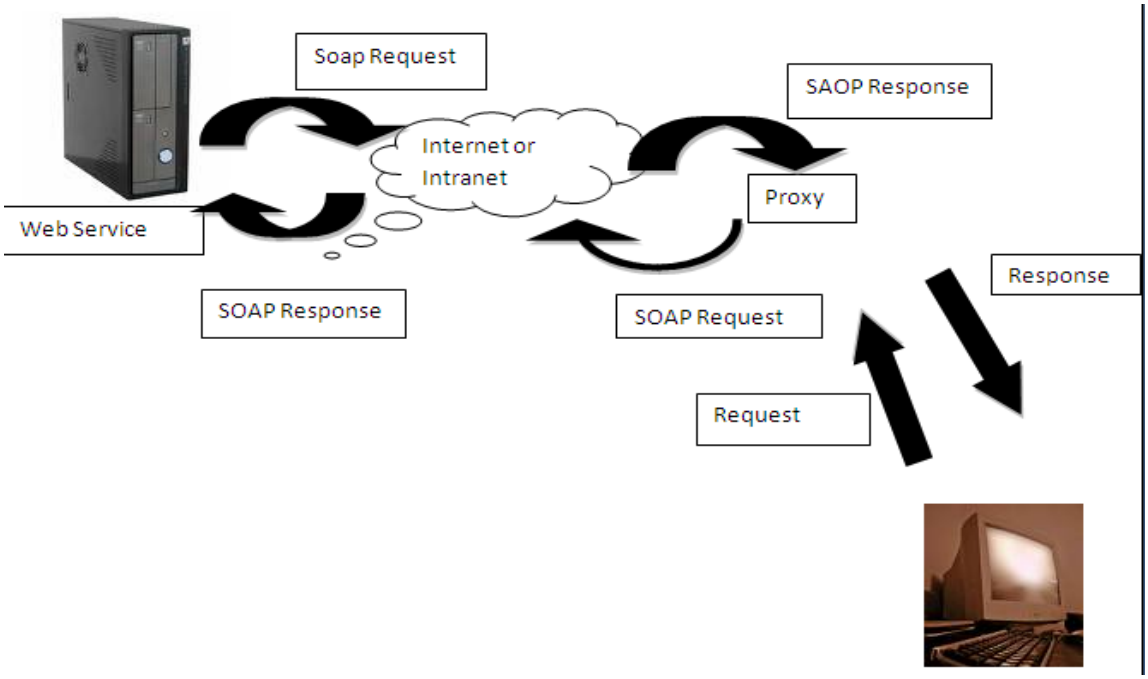


Figure 3.1 Proxy and SOAP in XML Web Service

Client applications send request to proxy and get response from proxy. It is an object-method relation. However proxy object should get flexible and readable messages. Communication between web services and clients should be design in common standards for using web services regularly. SOAP plays an important role in this situation. SOAP is a software protocol for exchange information between distributed environments using XML. It usually works over HTTP and also can be used in different

communication protocols such as FTP and SMTP. SOAP protocol is designed for exchanging information between Web Service and client applications in common standards. Microsoft designed SOAP Toolkit tool to monitor SOAP messages between clients and Web services.

SOAP consists of three parts:

- Envelope
- Data Encoding Rules
- Message Exchange Model
- Remote Procedure Call

3.1.2 SOAP Envelope

SOAP envelopes should be composed of header and body part. SOAP body includes information of request of methods in XML format. Figure 3.2 shows the structure of the envelope element.

Envelope is the root element of the SAOP XML message, and it is the most important part of SOAP message. Envelope element is required element of SOAP, others are optional.

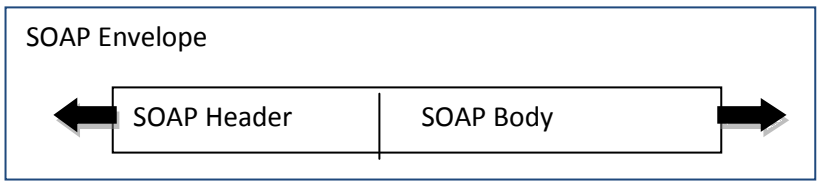


Figure 3.2 Structure of envelope

3.1.2.1 SOAP Body

Body element is the child element of the SOAP xml element. SOAP body contains the actual SOAP message in XML based format. Body element is composed of returned messages from web service and report errors.

3.1.3 SOAP Messages of Virtual Campus

SOAP request of Virtual Campus is sent to web service, and reference name is VCRSSService. VCRSSOperations has methods called MakeRSS and GetData. MakeRSS method takes dataset as an input parameter and create RSS file with related information.

Figure 3.3 shows the client request to web service MakeRSS method. It sends related dataset as an input parameter. SOAP messages are monitored by SOAP Toolkit. Figure 3.4 shows the response of the web service in XML format. Return parameter generate SOAP message body part.

This example shows the how client applications use the web service method by proxy object.

Figure 3.3 HTTP request

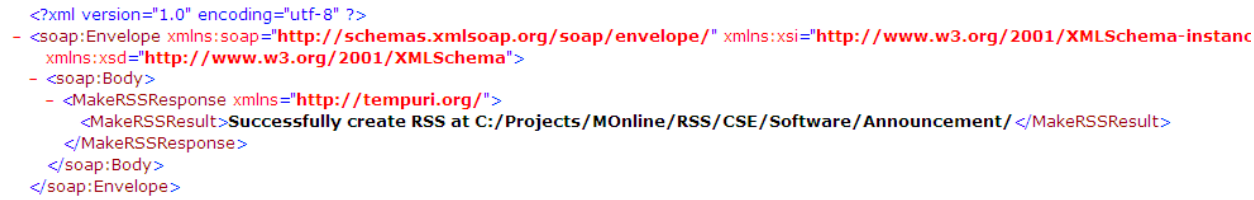


Figure 3.4 HTTP Response

3.2 Web Services

Web service is a software component. It is designed for the interoperaton of different web applications over internet or intranet. Web service is defined by W3C. Web services use the XML, SOAP, WSDL (Web Services Description Language) and UDDI (Universal Description, Discovery and Integration) open standards over an internet HTTP protocol. Figure 3.5 shows the web-based SOAP design. Web services are XML format to decode and encode data. SOAP protocol is used for transporting this data. It is used for accessing Web Service. WSDL describes the available web service and UDDI list the services that are available. Web services are mostly used for communicating and exchange information without any knowledge of each others' structure in business environment. All messages between client applications and services in XML format.

Also because of that it can use by any different operating system and application. Web services publish methods and object to other systems over internet. Other applications can also use these methods in their own system.

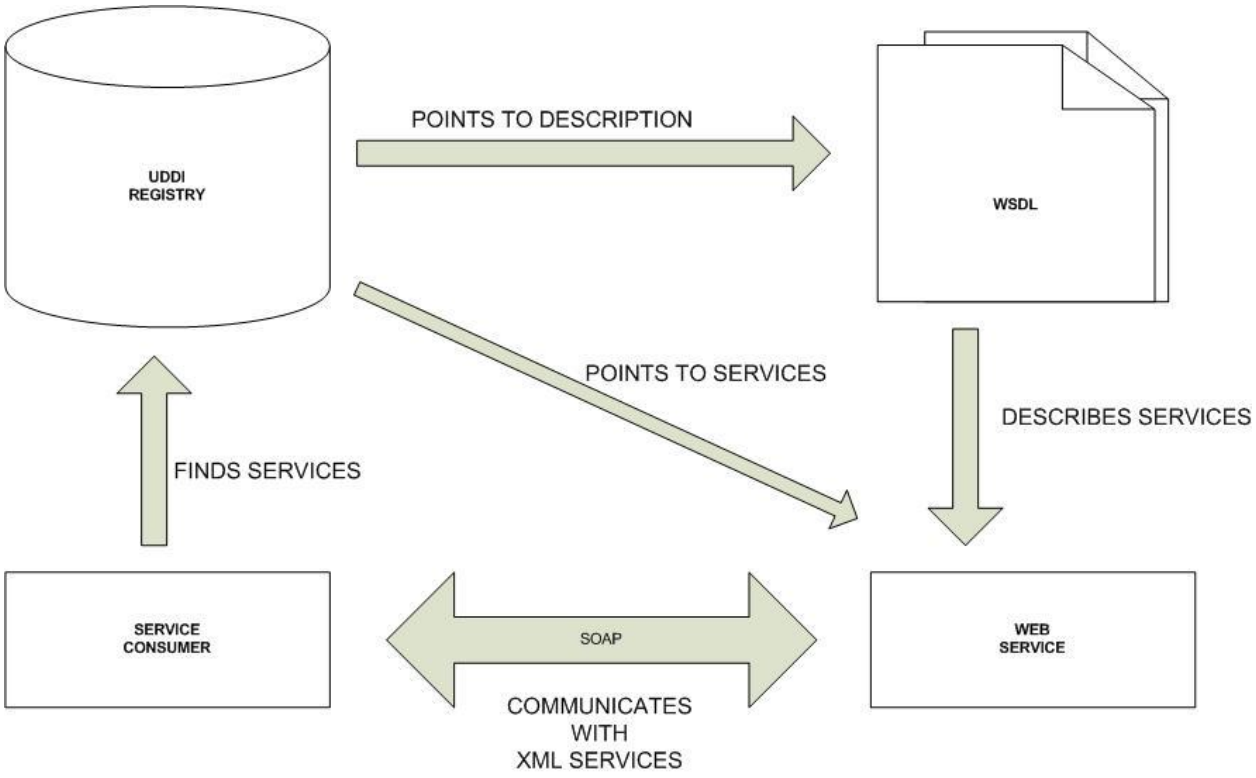


Figure 3.5 WEB-based and SOAP design

3.2.1 WSDL (Web Services Description Language)

WSDL is an interface of the web service. WSDL is a XML format for describing web service components, such as methods and properties. WSDL components are interfaces, bindings and services. .Net provides WSDL automatically create WSDL when adding a service reference to project. Simple WSDL document shown in Figure 3.6:

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitionsxmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"        targetNamespace="http://tempuri.org/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
 <wsdl:types>
  <s:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
   <s:element name="MakeRSS">
  <wsdl:message name="MakeRSSSoapIn">
  <wsdl:part name="parameters" element="tns:MakeRSS" />
 </wsdl:message>
 <wsdl:message name="MakeRSSSoapOut">
  <wsdl:part name="parameters" element="tns:MakeRSSResponse" />
 </wsdl:message>
 <wsdl:portType name="RSSOperationsSoap">
  <wsdl:operation name="MakeRSS">
   <wsdl:input message="tns:MakeRSSSoapIn" />
   <wsdl:output message="tns:MakeRSSSoapOut" />
  </wsdl:operation>
 </wsdl:portType>
 </wsdl:binding>
  <wsdl:service name="RSSOperations">
  <wsdl:port name="RSSOperationsSoap" binding="tns:RSSOperationsSoap">
   <soap:address location="http://localhost/VCRSSOperations/RSSOperations.asmx" />
  </wsdl:port>
  <wsdl:port name="RSSOperationsSoap12" binding="tns:RSSOperationsSoap12">
   <soap12:address location="http://localhost/VCRSSOperations/RSSOperations.asmx" />
  </wsdl:port>
 </wsdl:service>
</wsdl:definitions>
```

Figure 3.6 Sample WSDL Document in ZOI

WSDL descriptions are represented in XML-based language and show how to access Web service operations. WSDL uses elements to describe the Web Service. Main elements of Web Service are porttype, messages, types, and binding. <portType>

element holds the operations of the Web Service. <message> element include messages that used by service. <types> tag defines the used data types in web service. <binding> tag holds the protocols for communication used by web service. Web service also contains other elements such as extension elements and service elements.

3.3 RSS

RSS is a format to deliver regularly changing web contents. RSS is an abbreviation of Real Simple Syndication or Rich Site Summary. RSS is firstly developed by Netscape. Especially as blogs become widespread, RSS feed becomes a valid technology. If the Web page has RSS icon, you can easily access RSS content by using some RSS feed readers. Because RSS is an XML file format, it can be displayed on Internet Explorer 7 and Firefox. RSS is commonly used for share contents between web pages. RSS is most popularly used in broadcasting, blogs and podcasts. Furthermore it can be used in much different type of contents.

RSS allows internet users to easily follow information by retrieving the latest contents from the web sites. Users also save time by using RSS because they do not need to visit each site individually. Some of RSS feed readers or aggregator programs allow users to reach the RSS feeds from various sites and display their summary of contents for user. There are many RSS reader software developed for different platforms. Most popular RSS readers are Amphetadesk and FeedReaders. Amphetadesk is used in Windows, Linux and Mac. On the other hand, FeedReaders can be only used in Windows and NewsGater .It is designed for use of Outlook and windows integration. There are however some free readers such as Google Reader, My yahoo and Bloglines.

RSS usually contains description, title, and link for original web page. Figure 3.7 is an example of RSS feed. Main RSS tags are <rss> and <channel>.

```
<?xml version="1.0" encoding="utf-8"?>
<rss xmlns:a10="http://www.w3.org/2005/Atom" version="2.0">
    <channel>
        <title>CSE</title>
        <link>http://localhost/MOnline/Assignment/</link>
        <description>Materials Source</description>
        <language>tr-tr</language>
        <managingEditor>Selahattin.Kuru@isikun.edu.tr</managingEditor>
        <lastBuildDate>Thu, 04 Dec 2008 00:00:00 +0200</lastBuildDate>
        <category>Software</category>
        <item>
            <guid isPermaLink="false">1</guid>
            <link>http://localhost/MOnline/documents/assignments/CSE/Software/01/Homework/rss1.docx</link>
            <title>01</title><description>Homework1</description>
            <LastUpdatedDate>2008-12-01T00:00:00+02:00</LastUpdatedDate>
        </item>
    </channel>
</rss>
```

Figure 3.7 Sample RSS Feed

3.3.1 RSS Structure

RSS is a summary of web pages. The channel occurs on the top of the xml file and contains information that how items related to each other. RSS feed holds only one <channel> tag and it gives all the information about the web page. Channel tag contains child tags which are title, description, link, pubDate, language and category tags. Table 3.2 shows the child elements of channel tag. Title, link and description elements are required and rest of them are the optional elements.

Table 3.2 Child elements of channel tag

Element	**Description**	**Opt/Req**
Link	Contains link of real web page	Required
Description	Contains description of RSS feed	Required
Title	Defines title of RSS feed	Required
ManagingEditor	Responsible user e-mail of feed's content	Optional
lastBuildDate	Date of the feed's content	Optional
Category	Category of the feed	Optional
Language	Language of the RSS in written	Optional

Items of the channel tag

The name of the feed is written in <title> element. It is usually name of the web page. <link> tag holds the URL of the web page. <description> is the description of the feed. The <category> element is used for specifying the category of feed. Web pages can be grouped based on category fields. The category of RSS could be as shown:

<category>Software</category>

The <language> item specifies the language of html document. (in which the language is written). Example of the language could be as following:

<language>en-en</language>

The <managingEditor> is usually email of the user who is responsible of the feed content. (Webmaster) Example of the managingEditor could be:

<managingEditor >webmaster@course.com.tr< /managingEditor >

The <lastbuildDate> defines the last updated date of the web content. Example of the lastbuildDate could be:

< lastbuildDate >12-12-2008</ lastbuildDate >

The channel element has to contain at least one item tag.

Items of the <item> tag

Each item tag is a recently added content in a RSS. Required item elements in show

Table 3.3:

Table 3.3 Child elements of item tag

Element	**Description**	**Example**
title	The title of the item.	Venice Film Festival Tries to Quit Sinking
link	The URL of the item.	http://www.course.isikun.edu.tr
description	The item synopsis.	

Optionally other tags can be used optional. Item tag elements are shown in Table 3.4:

Table 3.4 Item tag elements

Element	**Description**
author	Email address of the author of the item. More.
category	Includes the item in one or more categories. More.
comments	URL of a page for comments relating to the item. More.
enclosure	Describes a media object that is attached to the item. More.
guid	A string that uniquely identifies the item. More.
pubDate	Indicates when the item was published. More.
source	The RSS channel that the item came from. More.

3.3.2 Data Providing

New types of classes which implements by .Net Framework 3.5, WCF (Windows Communication Foundation) allows syndication in RSS 2.0 format. RSS format has standards represent interoperability contents. As a result of this, clients can easily use this file in any platform. Interoperability information generally implemented as a Feed. Feed composed of author, title, URL and relational metadata elements. Furthermore it includes one or more item elements. Each item includes title, URL, description and category elements.

WCF is a programming framework used to build applications that intercommunicate. WCF is a part of the .Net framework dedicated to communications. WCF is designed in suitable with Service Oriented architecture principles to support distributed computing.

RSS is supported by WCF. WCF includes many classes' types such as SyndicationFeed, SyndicationItem, SyndicationLink, SyndicationPerson. WCF includes types for supporting different content sharing like RSS20FeedFormatter.

A WCF is a service that working on the enterprise content management system, that able to send information to authorized users by URL. URL information is an address like http://www.courseisikun.edu.tr/User=5 . After request of this URL, WCF service find related record which user id equals to 5 and then prepared this data in RSS format and send this data to end user in some way. Figure 3.8 show this flow:

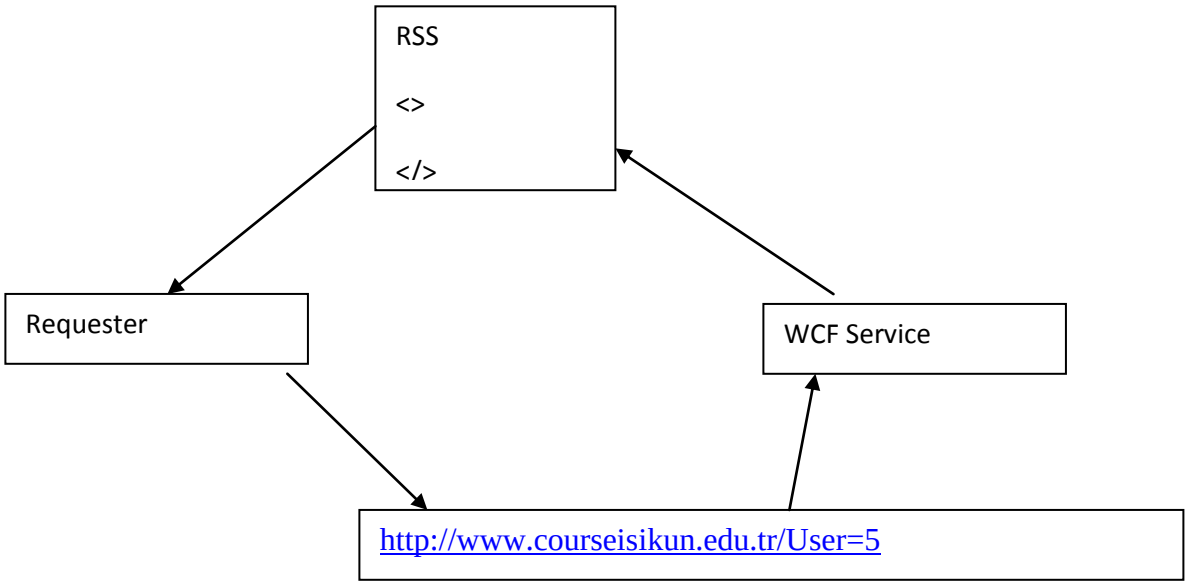


Figure 3.8 Flow of clients reach RSS URL

Hercule Course Online is composed by five different interfaces which are Assignments, Materials, Grades, Announcement and Web Resources. Although all of them have same

RSS structure, RSS tags fields contain different meanings. Each RSS tag holds different data. RSS feed should be designed in common standards. Other system administrator should have little knowledge about RSS and know RSS structure to use this information in their own system. RSS feeds indexing according to the student and use related information in system. The problem occurred here that the other system was not able to display the assignments and materials files in its own system. As a solution, these files were saved in the folders over RSS. Figure 3.9 shows the indexing algorithm. All parts of Hercule Course Online use the same algorithm for creating RSS feed. When the new record added in Admin Panel, existing RSS files are updated with valid records.

```
method MakeRSS(dataset)
 split dataset into data table
for each item in channel data table
        get  the “title”, “channel description”,”uri”  data to initialize feed
        add the <authors> to feed
        add the <category> to feed
        add the <language> to feed
        add the <link> to feed
        create channel items collection
        for each item in items data table
                if item belong to the channel // check with course name
                        add item into channel tag
                        add the <content> to item
                        add the <link> to item
                        add the <id> to item
                        add the <lastupdatedtime> to item
                end if
        end for
        add items to feed
        create RSS formatter with feed
        create xml in file system
        write rss to created xml
end for
```

Figure 3.9 Providing algorithm

MakeRSS method takes Dataset object as an input parameter. Dataset composed of two distinct data tables. First of all, dataset is splited into data tables. First data table includes channel data and second data table includes items data. For each data row in data tables has different data and they generated by relational stored procedure. A channel and Item data table includes data rows which represented in Table 3.5 and Table 3.6:

Table 3.5 Data rows of Channel data table in Hercule

Name	**Description**	**Example**
ChannelDescription	Defines the description of the channel	Material source
ManagingEditor	Defines the administrator email address	admin@isikun.edu.tr
CourseName	Defines the title of the channel	CSE
SubCourseName	Defines the category of the channel	Software
LastUpdatedTime	Defines the last updated time of the content	11-01-2009 23:00:00
Uri	Defines the link of the web content	http://www.courseonline.isikun.edu..tr
SubCourseId	Defines the id of the sub course which is generated by database	1
RssPath	Defines the RSS path in file system	C:\projects\Campus\RSS\

Table 3.6 Data rows of Item data table in Hercule

Name	**Description**	**Example**
Description	Defines the description of the item	Material source
ManagingEditor	Defines the administrator email address	admin@isikun.edu.tr
ItemTitle	Defines the title of the item	CSE
Id	Defines the item id of the item	1
LastUpdatedTime	Defines the last updated time of the content	11-01-2009 23:00:00
FilePath	Defines the link of the web content	http://www.courseonline.isikun.edu.tr/Assignments&AssignId=5
SubCourseId	Defines the id of the sub course which is generated by database	1

I have used SyndicationFeed class in .Net to create a feed. SyndicationFeed class represents a top-level feed object, <feed> and <rss> in RSS.

3.4 Scenario

1. Admin users define the course system parameters

- Define courses
- Define users and type of user

2. Instructor defines detail of course elements

- Define students
- Add students to related courses
- Adding materials
- Define announcements
- Define assignments
- Define students grades

3. When new features such as materials, assignments, grades and announcements are defined in a Hercule Course Online course system, other system resources will be updated with new information. Operations are follows:

- Authenticate to Hera Course Online and show resources
- Add new features in Hercule Course Online
- Again authenticate Hera Course Online to show new resources.

4. When new records define in Hercule Course Online system, it calls web service to create or update related RSS feed file. We can monitor these by Microsoft SOAP Toolkit.

- Open Microsoft SOAP toolkit
- Create new Trace file
- Monitor the HTTP request and response messages

Chapter 4

Technical Development of Virtual Campus

4.1 Development Environment

4.1.1 .Net Framework 3.5

The Microsoft .Net Framework is a software technology, developed by Microsoft, and designed for application development platform. It supports Web based and Windows applications. The .Net includes CLR(Common Language Runtime) and the .Net Framework class library. The CLR is runtime environment for development applications and .Net Framework library provides standard implementations of many services for CLR.

4.1.2 C#

C# is a programming language and is developed by Microsoft. It is an object-oriented, simple and modern language. C# is powerful and flexible programming language; it can be used for developing various types of applications.

4.1.3 SQL Server 2005

SQL server 2005 is a of relational database management software. It provides create and manage databases. It is one of the most powerful and secure database server. It allows remote connections from clients to manage data.

4.2 User Types of Virtual Campus

4.2.1 User Types

Three types of users are defined in Virtual Campus for accessing system. They are administrator, instructor and students. Admin user can access all part of the systems. Administrator has read, modify, write, and access privileges on all operations. Admin users can define courses, sub courses, assignments, materials, and users in system.
Instructor account can access the admin panel of the system. Instructors are defined by administrators. They can only define features according to the defined courses. They can define materials, assignments, announcements, students' grades and courses.

Students only have read privileges and cannot access administrator panel. They access course system. They can only reach their own course materials and open courses for every student.

4.3 Software Architecture of Virtual Campus

4. 3.1 n-Tier Architecture

First step of development is deciding the architecture of an application. Enterprise application contains complex business logic and as a result of this, the complexity of code is increasing. N tier architecture design is for improving quality of implementation and reducing complexity of code. Various issues have to be considered while design of architecture process, such as scalability, reusability, maintainability and security. N-tier architecture means that the solution splits into different projects. Each project can be located on different servers. Therefore, if one layer should be change, you do not have to compile the whole solution. For instance, if something has been changed in database,

you just only change data access layer. In 3-tier architecture there are 3 different layers which are presentation layer, business layer and data access layer. Each layer of solution has different responsibilities. Sample example of 3-tier project in Figure 4.1.

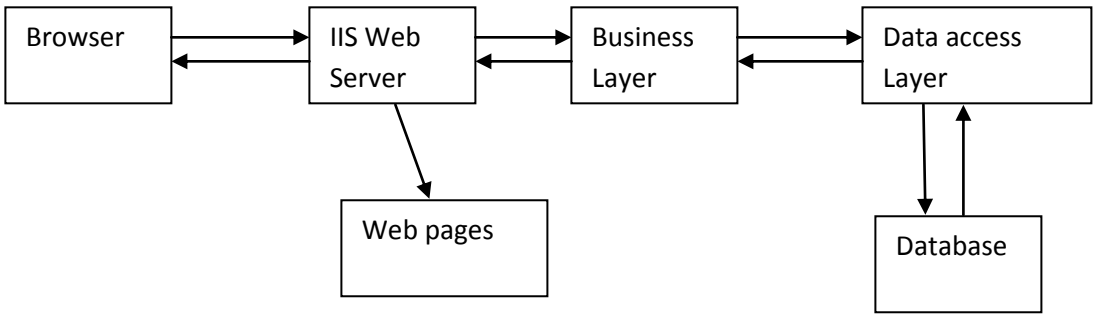


Figure 4.1 Architecture design of Virtual Campus

Virtual Campus has been implemented by 3-tier architecture. In Virtual Campus, Hercule Course Online, Hera Course Online and Admin Panel represent the Presentation layer. They are web project. Business layer and data access layer.

Presentation Layer

Presentation layer is the client side layer. It is designed for end user data manipulation. Presentation layer is concerned with interacting with the user and display data to user. Presentation layer consists of web pages or windows forms. Presentation layer is developed by web pages in Virtual Campus.

Course Define

Admin user of the Course Online System defines the courses information. Screenshot of Course Define is shown in Figure 4.2:

Figure 4.2 Screenshot of Admin Course define

Sub Course Define

Admin user of the Course Online System defines the sub courses information. Screenshot of Course Define is shown in Figure 4.3:

Figure 4.3 Screenshot of Admin Sub Course define

User Define

Admin user of the Course Online System defines the users' information. Screenshot of User Define is shown in Figure 4.4:

Figure 4.4 Screenshot of Admin User define

User Courses Define

Instructor of the Course Online System defines the users' courses information. Screenshot of User Courses is shown in Figure 4.5:

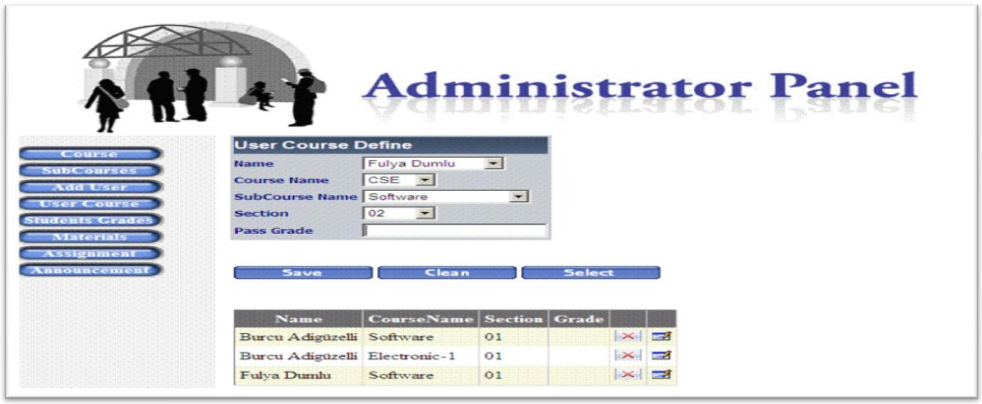


Figure 4.5 Screenshot of Admin Users' Course define

Students Grades Define

Instructor of the Course Online System defines the users' grades information. Screenshot of User grade is shown in Figure 4.6:

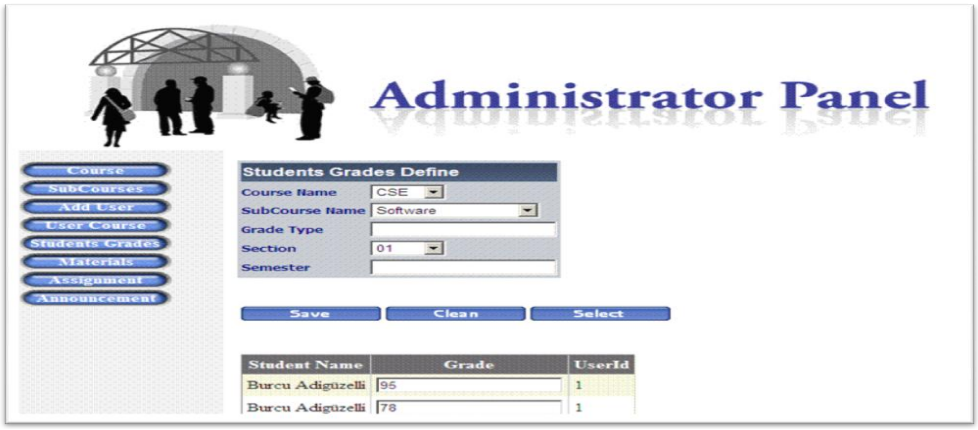


Figure 4.6 Screenshot of Admin Students' Grades define

Materials Define

Instructor of the Course Online System defines the course materials information. Screenshot of Course Materials is shown in Figure 4.7:

Figure 4.7 Screenshot of Admin course Materials define

Assignments Define

Instructor of the Course Online System defines the course assignments information. Screenshot of Course Materials is shown in Figure 4.8:

Figure 4.8 Screenshot of Admin course Assignments define

Announcement Define

Instructor of the Course Online System defines the course assignments information. Screenshot of Course Announcement is shown in Figure 4.8:

Figure 4.9 Screenshot of Admin course Announcement define

Hercule Course Online Login

Hercule Course Online Login page is shown in Figure 4.10:

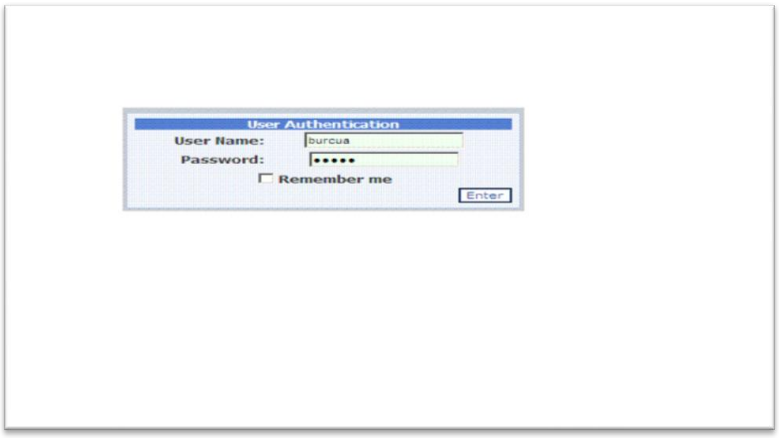


Figure 4.10 Hercule Course Online Login page

Hercule Course Online Assignments

Hercule Course Online Assignment page is shown in Figure 4.11:

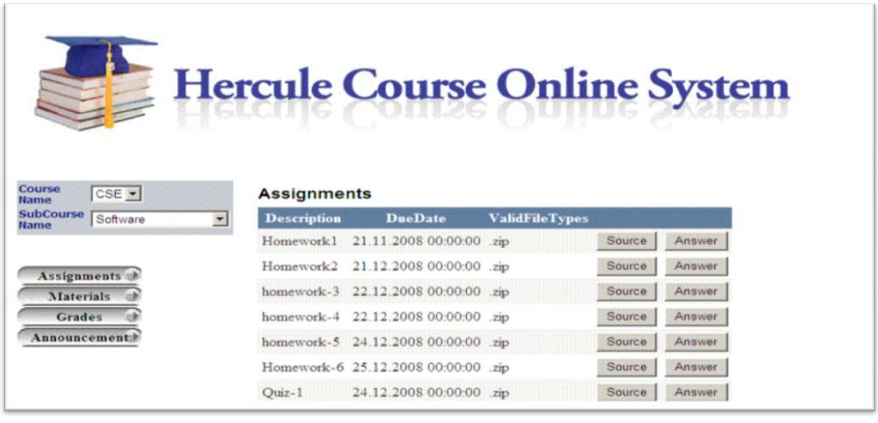


Figure 4.11 Hercule Course Online Assignment page

Hercule Course Online Announcement

Hercule Course Online Announcement page is shown in Figure 4.12:

Figure 4.12 Hercule Course Online Announcement page

Hercule Course Online Materials

Hercule Course Online Materials page is shown in Figure 4.13:

Figure 4.13 Hercule Course Online Materials page

Hercule Course Online Grades

Hercule Course Online Grades page is shown in Figure 4.14:

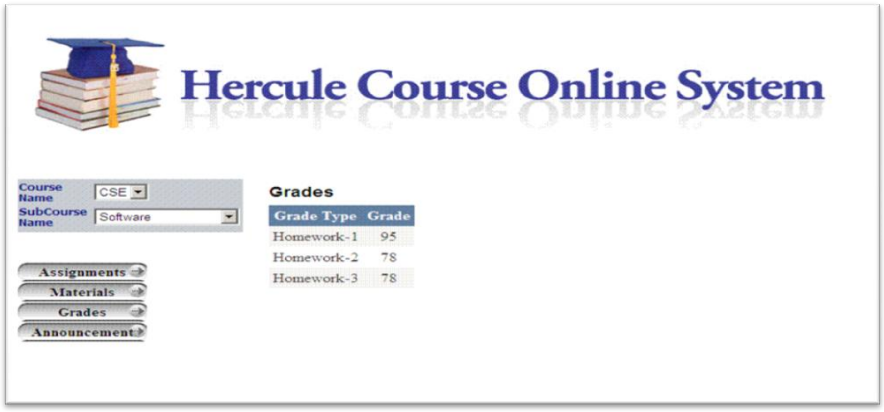


Figure 4.14 Hercule Course Online Grades page

Hera Course Online Assignments

Hera Course Online Assignment page is shown in Figure 4.15:

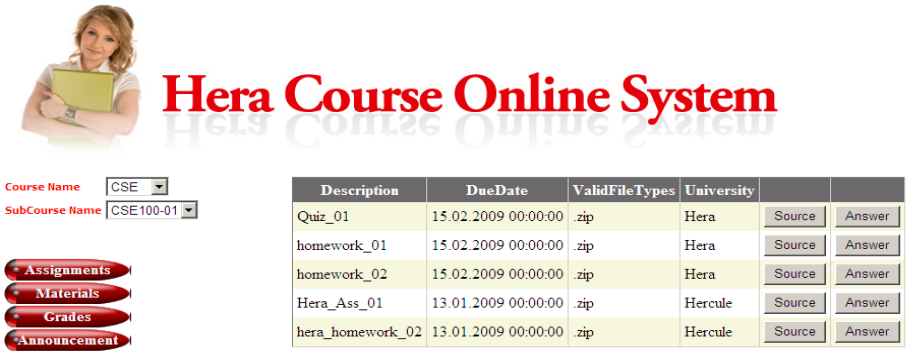


Figure 4.15 Hera Course Online Assignment page

Hera Course Online Materials

Hera Course Online Materials page is shown in Figure 4.16:

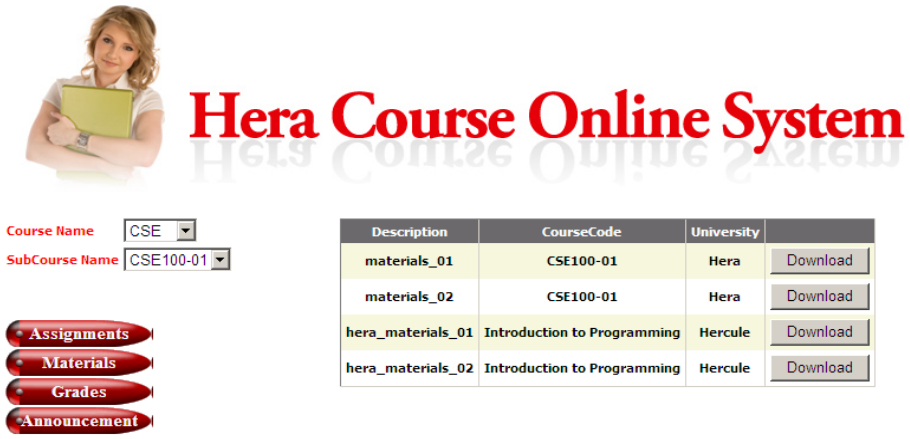


Figure 4.16 Hera Course Online Materials page

Hera Course Online Grades

Hera Course Online Grades page is shown in Figure 4.17:

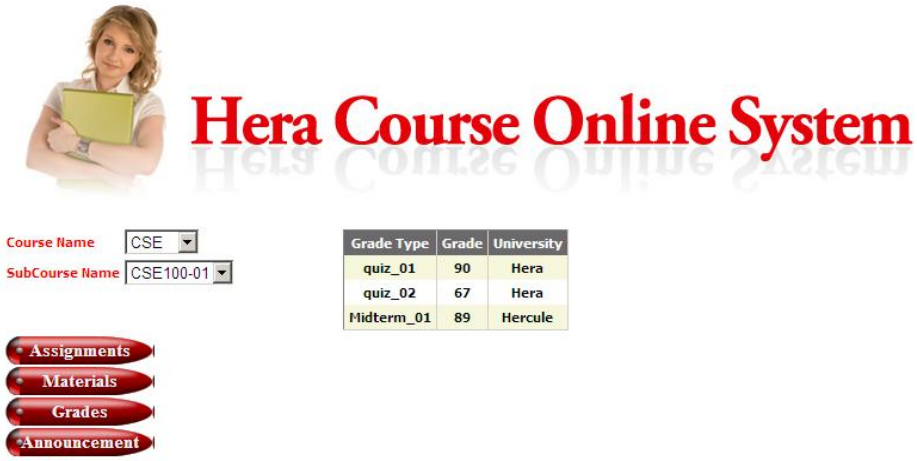


Figure 4.17 Hera Course Online Grades page

Hera Course Online Announcement

Hera Course Online Announcement page is shown in Figure 4.18:

Figure 4.18 Hera Course Online Announcement page

Business Layer

Business layer is a bridge between presentation layer and data access layer. It processes the requests of the end users. It does not store the data. Business layer is responsible for processing the data retrieved from user and send this data to data access layer. It has a reference to the data access layer. It connects to the data access layer whenever it needs data.

Data access Layer

Data access layer (DAL) is responsible for accessing database. DAL composed of collection of classes, interfaces and their methods. Data access layer connects to the relational database via OleDBConnection and return data in datasets. Only DAL has right to access to database. Other layers do not have any information about database.

DAL is designed for database security. Only DAL keep stored procedure names, parameters and connection string hard coded and other layers do not know sensitive information about database. Because of using SP increase the performance and it is more secure than SQL text strings, only stored procedures are used in Virtual Campus. DAL deals with data manipulation actions such as Insert, Delete, Update and Select. Database can be SQL, MySQL, Oracle, Access and XML. SQL Server 2005 is used in Virtual Campus.

Database Layer

Database layer of Virtual Campus includes two different databases. They have same database structure but have different records. Databases include 8 tables. Each table of database have foreign-key relation. Relation of tables are shown in Figure 4.19:

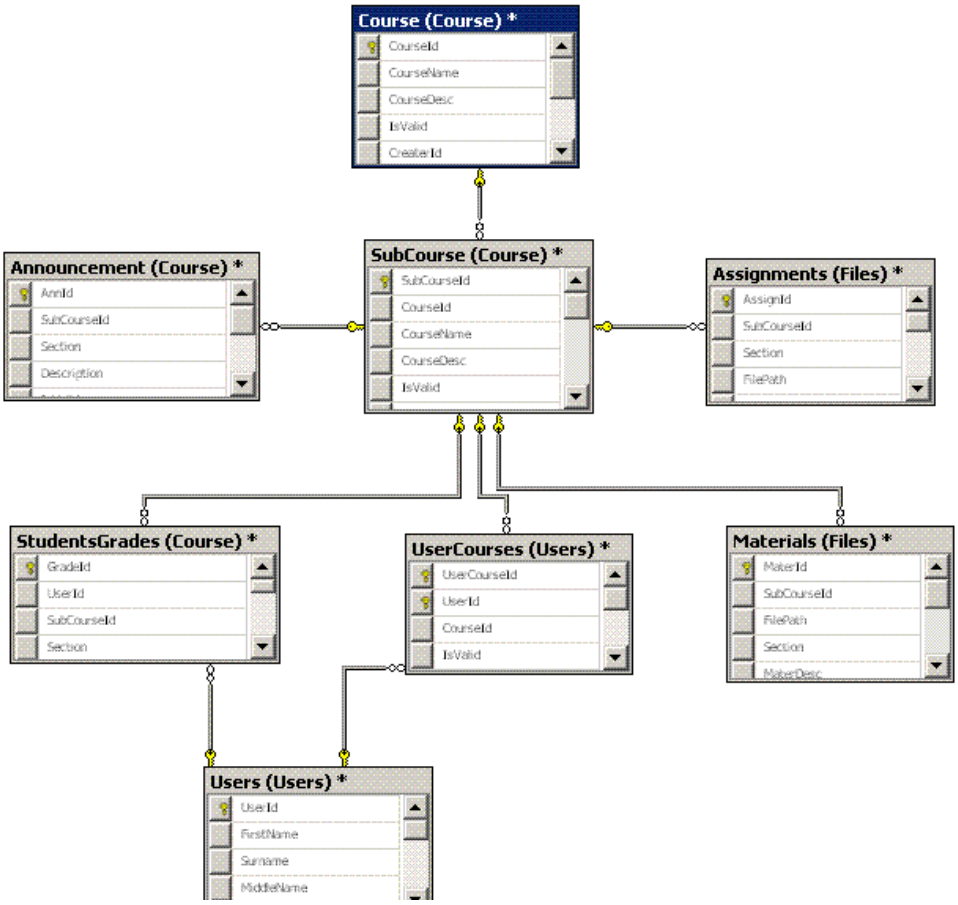


Figure 4.19 Relational Database Diagram

Fact table of the system is Course table. All tables are having “IsValid”, “CreaterId”, “CreatedDate”, “ModifierId” and ModifiedDate columns. IsValid is holds the record is valid or deleted. CreaterId holds the userid who is adding the record. CreatedDate defines the date of the record created. It is get by system date and use getdate() method. ModifierId also holds the userid who is last updated a data. ModifiedDate holds the date of the updated time. Database structure is shown below:

Database Structure

Announcement

Announcement table holds the detail data of announcement. The structure of each columns and properties in courses table is shown in Figure 4.20:

Column Name	Data Type	Allow Nulls
AnnId	int	☐
SubCourseId	int	☐
Section	char(2)	☑
Description	nvarchar(250)	☑
IsValid	char(1)	☑
CreaterId	int	☑
CreatedDate	datetime	☑
ModifierId	int	☑
ModifiedDate	datetime	☑
		☐

Figure 4.20 Structure of Announcement table

Assignment

Assignment table holds the detail data of assignments. The structure of each columns and properties in courses table is shown in Figure 4.21:

Column Name	Data Type	Allow Nulls
AssignId	int	☐
SubCourseId	int	☐
Section	char(2)	☑
FilePath	nvarchar(250)	☑
AssignDesc	nvarchar(250)	☑
IsValid	char(1)	☑
CreaterId	int	☑
CreatedDate	datetime	☑
ModifierId	int	☑
ModifiedDate	datetime	☑
DueDate	datetime	☑
ValidFileTypes	char(5)	☑
AnswersPath	nvarchar(250)	☑
		☐

Figure 4.21 Structure of Assignment table

Course

Courses table holds the detail data of courses. The structure of each columns and properties in courses table is shown in Figure 4.22:

Column Name	Data Type	Allow Nulls
CourseId	int	☐
CourseName	nvarchar(250)	☑
CourseDesc	nvarchar(250)	☑
IsValid	char(1)	☑
CreaterId	int	☑
CreatedDate	datetime	☑
ModifierId	int	☑
ModifiedDate	datetime	☑
		☐

Figure 4.22 Structure of Course table

Course

Courses table holds the detail data of courses. The structure of each columns and properties in courses table is shown in Figure 4.23:

	Column Name	Data Type	Allow Nulls
🔑	MaterId	int	☐
	SubCourseId	int	☐
	FilePath	nvarchar(250)	☑
	Section	char(2)	☑
	MaterDesc	nvarchar(250)	☑
	IsValid	char(1)	☑
	CreaterId	int	☑
	CreatedDate	datetime	☑
	ModifierId	int	☑
	ModifiedDate	datetime	☑
			☐

Figure 4.23 Structure of Materials table

Course

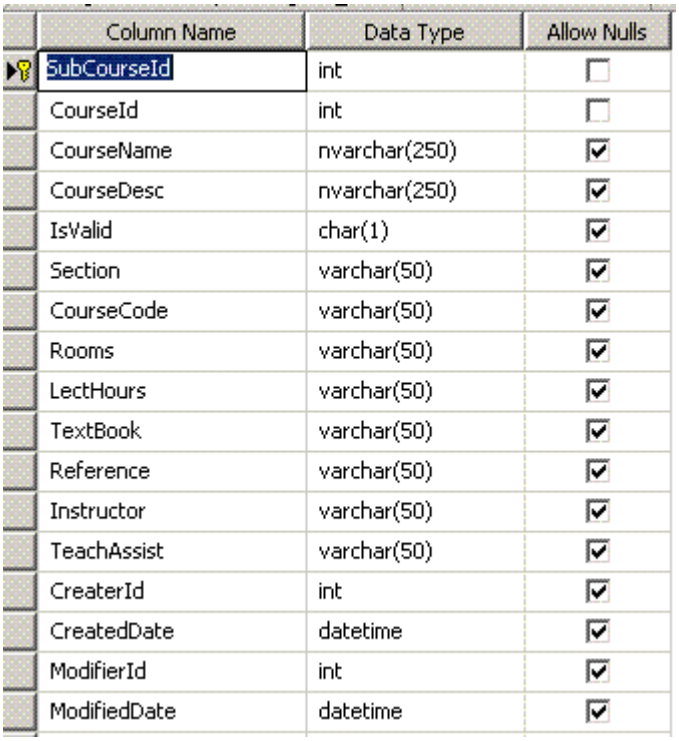
UsersGrades table holds the detail data of users' grades. The structure of each columns and properties in courses table is shown in Figure 4.24:

	Column Name	Data Type	Allow Nulls
🔑	GradeId	int	☐
	UserId	int	☐
	SubCourseId	int	☐
	Section	varchar(50)	☑
	GradeType	varchar(50)	☑
	Grade	char(2)	☑
	Semester	varchar(20)	☑
	IsValid	char(1)	☑
	CreaterId	int	☑
	CreatedDate	datetime	☑
	ModifierId	int	☑
	ModifiedDate	datetime	☑
			☐

Figure 4.24 Structure of Users Grades table

SubCourse

SubCourse table holds the detail data of sub courses. The structure of each columns and properties in courses table is shown in Figure 4.25:



Column Name	Data Type	Allow Nulls
SubCourseId	int	☐
CourseId	int	☐
CourseName	nvarchar(250)	☑
CourseDesc	nvarchar(250)	☑
IsValid	char(1)	☑
Section	varchar(50)	☑
CourseCode	varchar(50)	☑
Rooms	varchar(50)	☑
LectHours	varchar(50)	☑
TextBook	varchar(50)	☑
Reference	varchar(50)	☑
Instructor	varchar(50)	☑
TeachAssist	varchar(50)	☑
CreaterId	int	☑
CreatedDate	datetime	☑
ModifierId	int	☑
ModifiedDate	datetime	☑

Figure 4.25 Structure of Sub Courses Grades table

Users Course

Courses table holds the detail data of courses. The structure of each columns and properties in courses table is shown in Figure 4.26:

Column Name	Data Type	Allow Nulls
UserCourseId	int	☐
UserId	int	☐
CourseId	int	☐
IsValid	char(1)	☑
Section	varchar(50)	☑
Grade	varchar(50)	☑
CreaterId	int	☑
CreatedDate	datetime	☑
ModifierId	int	☑
ModifiedDate	datetime	☑
		☐

Figure 4.26 Structure of Users Courses table

Course

User table holds the detail data of users. The structure of each columns and properties in users table is shown in Figure 4.27:

Column Name	Data Type	Allow Nulls
UserId	int	☐
FirstName	nvarchar(50)	☑
Surname	nvarchar(50)	☑
MiddleName	nvarchar(50)	☑
UserName	nvarchar(50)	☑
Password	nvarchar(50)	☑
Email	nvarchar(50)	☑
UserType	int	☑
IsValid	char(1)	☑
CreatedDate	datetime	☑
CreaterId	int	☑
ModifiedDate	datetime	☑
ModifierId	int	☑
		☐

Figure 4.27 Structure of Users table

Chapter 5

Conclusion

In this thesis, design of a VO platform has been implemented for course systems. Nevertheless it is able to use at different business environment with some modification. This design addressed the problems of aggregation of heterogonous systems and security.

Machine-to-machine strategy plays an important role in design and implementation of ZOI. The technologies of Web services, RSS Feed and SOAP protocol, and interoperability have been used in development of system.

References

1. Song,W., Zhou,M., "A Contextual Based Semantic Modeling Approach to Task-Service Formation in Virtual Organization", *International Conference on Web Intelligence*, 2006

2. Robinson P., Karabulut Y., Haller J., "Dynamic Virtual Organization Management for Service Oriented Enterprise Applications", *IEEE*, 2005

3. Zhou L., Chen H., Mao Y., "A Semantic-based Framework for Virtual Organization Management", *IEEE*, 2008

4. Li J., Huai J., Hu C., "PEACE-VO: A Secure Policy-Enabled Collaboration Framework for Virtual Organizations", *IEEE*, 2007

5. Ferraiolo D., Sandhu R., Gavrila S., "The NIST Model for Role-Based Access Control", *Towards A Unified Standard*,2005

6. Mowshowitz A., "On the Theory of Virtual Organization", *Willey InterScience*, 1997

7. Lin A., Vullings E., Dalziel J., "A Trust-based Access Control Model for Virtual Organizations", *International Conference on Grid and Cooperative Computing Workshops*, 2006

8. Christopher P. Lockett G., "Business Trust and the Formation of Virtual Organizations", *IEEE*, 1998

9. Katzy B., "Design and Implementation of Virtual Organizations", *IEEE*, 2006

10. Lee Y., "A Dynamic Virtual Organization Solution for Web-Services Based Grid Middleware", *International Workshop on Database and Expert Systems Applications*, 2005

11. Jun T., Ji-chang S., Hui W., "Research on the Evolution Model for the Virtual Organization of Cooperative Information Storage", *International Conference on Management Science & Engineering (14th)*, 2007

12. Niinimaki M., White J., Cerff W., Hahkala J., Niemi T., Pitkanen M., "Using Virtual Organizations Membership System with EDG's Grid Security and Database Access", *International Workshop on Database and Expert Systems Applications,*

13. Crowston, K, Scozzi, B.," Open source software projects as virtual

14. Organizations Competency rallying for software development", *IEEE,* 2002

References Not Citied

Darko-Ampem S, Katsou M., "Secure Negotiation in Virtual Organizations, IEEE, 2006

Grechanik, M., and Conroy, K M., "Composing Integrated Systems Using GUI-Based Applications and Web Services", *IEEE International Conference on Services Computing*, 2007

Pohl, K., "A Translational Web Services Bridge for Meaningful Interoperability Among Information Systems", *IEEE*, 2006

Athanasopoulos G., Tsalgatidou A., Pantazoglou M., "Interoperability among Heterogeneous Services", *International Conference on Services Computing*, 2006

Liang P., He K., Li B., Liu J., "The Interoperability between Different e-Business Specifications, International Conference on Information Technology: Coding and Computing, Thang LE DINH, Tang Ho LE, Towards an Approach for Modeling Interoperability of Information Systems", *IEEE*, 2007

Zhou Y. Chen X., Wang C. , "A Self-Organizing Search Engine for RSS Syndicated Web Contents"**,** *International Conference on Data Engineering Workshops*, 2006

Chmielewski D., Hu G., "A Distributed Platform for Archiving and Retrieving RSS Feeds", *International Conference on Computer and Information Science*, 2005

http://www.w3.org/TR/2000/NOTE-SOAP-20000508/ , Simple Object Access Protocol (SOAP) 1.1

http://www.w3schools.com/soap/ , SOAP Tutorial

http://www.w3schools.com/rss/default.asp , RSS Tutorial

http://www.feedforall.com/industry-solutions.htm, Industry Solution

http://www.computorcompanion.com/LPMArticle.asp?ID=194, Write Your Own .NET RSS Feed in C#

http://www.dotnetspider.com/resources/1566-n-Tier-Architecture-Asp.aspx, n - Tier Architecture in Asp.net

http://www.pha.jhu.edu/~mmathis/seminar/ntier.html, The N-tier Architecture

http://jcmc.indiana.edu/vol3/issue4/desanctis.html#Virtual, Communication Processes for Virtual Organizations

http://www.w3.org/TR/ws-arch/, Web Services Architecture

http://www.webopedia.com/TERM/W/Web_services.html, Web services

Appendix A

A1. CD includes the source code, soft copy of the thesis in .pdf types and presentation.

Curriculum Vitae