

# IS 690 Web Services

## Base Syllabus

October 10, 2010

### Faculty Coordinator

Dr. Min Song

### Overview

This course provides a practical approach to develop web services for web-based information systems. Topics include (1) Service-oriented Architecture (SOA); (2) Web Services Models such as [Business Process Modeling Language](#) (BPML); (3) [Security and Authorization](#) for Web Services; (4) Web Services Repository such as [Universal Description, Discovery, and Integration](#) (UDDI); (5) Web Services Messaging such as Simple Object Access Protocol (SOAP) and [Representational State Transfer](#) (REST);

### Objectives

The goals of this course are to:

- Provide students with an opportunity to implement a network based application modeled in terms of a Service Oriented Architecture.
- Explore the new kinds of capability that may be developed and deployed using loosely coupled services with published interfaces and dynamic composition.
- Provide students with a hands-on experience with the full range of technologies that support service oriented approaches.
- Provide students with an opportunity to reflect on the differences between traditional client server, SOA, and Web service approaches to systems.

### Material that Must be Covered during the Semester

- Understanding SOA and Web Services
- Services, Descriptions, and Messaging
- Advanced Messaging and Metadata
- [Security and Authorization](#)
- Service Orientation Principles
- Application and Business Layers
- Service Orientated Analysis
- Web Services Models – BPML, Model-driven Architecture (MDA)
- SOA Design – WSDL
- SOAP
- REST
- UDDI

### Prerequisite

IS631 and CS602 or equivalent; or permission of the instructor

## Possible Textbook

Thomas Erl, *Service-Oriented Architecture (SOA): Concepts, Technology, and Design*, Prentice Hall PTR (August 2, 2005). [ISBN: 0131858580]

## Grading Scheme

Grades are assigned based on 3 assignments, midterm, a final project, and class participation. The grading breakdown is as follows:

- \* Assignments: 20%
- \* Midterm: 20%
- \* Final Project: 45%
- \* Class Participation: 15%

## Schedule

	TOPIC	TEXTBOOK READING
Week 1	<b>Class Overview</b> <b>Introduction to Web Services</b>	Chapter 1
Week 2	<b>Service-oriented Architecture</b> Principles of Service-Orientation	Chapter 5
Week 3	<b>Web Services and XML</b>	Chapter 3
Week 4	<b>Web Services Description Language</b> WSDL	Chapter 2
Week 5	<b>Web Services Models</b> BPML	Chapter 7
Week 6	<b>Web Services Messaging I</b> SOAP	
Week 7	<b>Web Services Messaging II</b> RESTful Web Services	Chapter 7 and Chapter 8
Week 8	<b>Midterm</b>	Chapter 9
Week 9	<b>Web Services Repository</b> UDDI	Chapter 13
Week 10	<b>Principles of Service-Orientation</b>	Chapter 8
Week 11	<b>Security and Authentication</b>	Chapter 9
Week 12	<b>Middleware</b>	Chapter 14
Week 13	<b>Meta Data and Interoperability between Different Information Resources</b>	Chapter 14
Week 14	<b>Future of Web Services</b>	
Week 15	<b>Final Project Presentations</b>	

## Sample Homework Questions

**Developing web services with AXIS2**

/\*\*

\* The service implementation class

```

*/
public class SimpleService {
/**
 * The echo method which will be exposed as the
 * echo operation of the web service
 */
    public String echo(String value) {
        return "from web services" + value;
    }
}

```

Each Axis2 service must have a services.xml file which will inform Axis2 about the service.

Following is the services.xml file contents for the SimpleService Web service:

```

<service>
    <parameter name="ServiceClass" locked="false">
        SimpleService</parameter>
    <operation name="echo">
        <messageReceiver
class="org.apache.axis2.rpc.receivers.RPCMessageReceiver"/>
    </operation>
</service>

```

Axis2 expects services to be packaged in a certain format. Let's do that now.

- a. export JAVA\_HOME=your java\_home
- b. export PATH=\$JAVA\_HOME/bin:\$PATH
- c. mkdir temp
- d. javac SimpleService.java -d temp/
- e. mkdir temp/META-INF
- f. cp services.xml temp/META-INF/
- g. cd temp
- h. jar -cvf SimpleService.aar \*

There are two primary hosting environments available for AXIS2 services: [1] Using the SimpleHTTPServer that is available in the Apache Axis2 distribution, or [2] Using Axis2 with Tomcat. You choose either one of the options.

### Sample Semester Group Project

You will work as a group (up to 3 people). Once you form a group, you have to specify the part done by each member of the group both in your proposal and in your term report. Every team must submit a proposal. The proposal should include the title, description, significance, hardware, and software involved, the procedure to work on, and the items to be delivered at the end of the project.

## **I. Overview**

### Service-Oriented Architecture (SOA) Business Services Platform

An organization creates a service-oriented architecture (SOA) to interface business functions among multiple application groups within the organization. A classic example of this is the credit/debit business processes required in banking institutions. Deposits and withdrawals from multiple sources may occur for a single account. Data from these transactions must be interfaced and coordinated so that the account balance is accurate.

SOAs use Web service technologies, such as Simple Object Access Protocol (SOAP) and WSDL definitions, to integrate applications using standard, reusable interfaces with low development, deployment and maintenance costs. XML often is used to transport the data between systems.

## **II. Project Description**

### *Deliverable*

The student team shall develop a SOA, enabled by Web services technologies, to access data from various back office databases and systems. The team will analyze and model the sales and financial business services of the client organization and then design a SOA solution.

The team will develop, test and implement a simple proof-of-concept business function to validate that the SOA is technically sound and can be scaled across an organization.

Either open source technologies or off-the-shelf technologies from Microsoft or IBM can be used for the solution.

### *Methods*

The software process will involve:

- Requirements gathering from the client
- Business analysis of requirements
- Development
- Quality assurance testing
- Deployment of the application
- Project management tracking

Teams shall use an iterative or spiral development process and employ Agile methodologies where appropriate. Teams shall interact directly with a client liaison regarding project status, requirements, etc.