

**School of Computing and Information Sciences**

**Course Title:** Component-Based Software Development

**Date:** Sept 8, 2010

**Course Number: COP 4814**

**Number of Credits: 3**

<b>Subject Area:</b> Programming	<b>Subject Area Coordinator:</b> Kip Irvine <b>Email:</b> irvinek@cs.fiu.edu
<b>Catalog Description:</b> Integrating, exchanging, and transforming XML data, building software from components, understanding security concepts, basic Web services.	
<b>Textbooks:</b> XSLT, 2 <sup>nd</sup> Edition, by Doug Tidwell. O'Reilly, 2008 Enterprise JavaBeans 3.1, by Andrew Lee Rubinger and Bill Burke (2010, O'Reilly)	
<b>References:</b>	
<b>Prerequisites Courses:</b> CGS 4366, CGS 4854	
<b>Corequisites Courses:</b>	

Type: Required for IT major

Prerequisites Topics:

- Familiarity with style sheets, XHTML
- Experience with querying and updating databases
- Knowledge of creating web applications
- Experience with simple data validation techniques
- Knowledge of object-oriented programming

Course Outcomes:

1. Be familiar with the principles of integrating and exchanging data
2. Master the use of XML/DOM to integrate and exchange data, and XSL/XSLT to transform data
3. Be familiar with principles for building software systems from components, including service-oriented architecture and appropriate design patterns
4. Be familiar with evidence-based security and code access security
5. Be familiar with the functions of a software version control system
6. Be familiar with building server applications that consume web services, exchange, and transform web-based data

**School of Computing and Information Sciences**  
**COP 4814**  
**Component-Based Software Development**

**Relationship between Course Outcomes and Program Outcomes**

<b>BS in IT: Program Outcomes</b>	<b>Course Outcomes</b>
a) Demonstrate practical hands-on expertise in selection, installation, customizing and maintenance of the state-of-the-art computing infrastructure.	
b) Demonstrate practical proficiency in selection, installation, customizing and maintenance of the state-of-the-art software systems.	
c) Demonstrate general understanding of at least one field where Information Technology plays a central role.	1, 2, 3, 4, 5, 6
d) Demonstrate understanding of the social and ethical concerns of the practice of Information Technology.	
e) Demonstrate the ability to work cooperatively in teams.	
f) Demonstrate effective communication skills.	
g) Demonstrate familiarity with fundamental ideas and issues in the arts, humanities and social sciences.	

**Assessment Plan for the Course & how Data in the Course are used to assess Program Outcomes**

Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan:

<http://www.cis.fiu.edu/programs/undergrad/it/assessment/>

## Outline

Topic	Number of Lecture Hours	Outcome
<ul style="list-style-type: none"> <li>• Data Representation and encoding               <ul style="list-style-type: none"> <li>○ ASCII, Unicode</li> <li>○ Binary, Octal, Hexadecimal</li> </ul> </li> </ul>	2	1
<ul style="list-style-type: none"> <li>• XML Documents               <ul style="list-style-type: none"> <li>○ Metadata</li> <li>○ XML namespaces</li> <li>○ Document Type Definition (DTD)</li> <li>○ Querying documents using XPath</li> </ul> </li> </ul>	6	1,5
<ul style="list-style-type: none"> <li>• XML Translation               <ul style="list-style-type: none"> <li>○ XML schemas</li> <li>○ XSLT template-based transformation</li> <li>○ XSL files</li> <li>○ translating XML to XHTML</li> <li>○ Selecting XML data with XPath</li> </ul> </li> </ul>	6	1,5
<ul style="list-style-type: none"> <li>• Integrative Coding               <ul style="list-style-type: none"> <li>○ Interfaces (review)</li> <li>○ Facade, Decorator design patterns</li> <li>○ Factory methods</li> </ul> </li> </ul>	6	2
<ul style="list-style-type: none"> <li>• Overview of Architectures for Integrating Systems               <ul style="list-style-type: none"> <li>○ DCOM</li> <li>○ CORBA</li> <li>○ RMI</li> <li>○ .NET</li> </ul> </li> </ul>	3	2

<ul style="list-style-type: none"> <li>• Web Services Overview <ul style="list-style-type: none"> <li>○ Applications of Web services</li> <li>○ SOAP, UDDI, WSDL</li> <li>○ Consuming Web services</li> </ul> </li> </ul>	4	2,5
<ul style="list-style-type: none"> <li>• Security Concepts <ul style="list-style-type: none"> <li>○ evidence-based security</li> <li>○ code access security</li> <li>○ overview of best security practices</li> <li>○ authentication to system resources and services</li> <li>○ encrypting data</li> </ul> </li> </ul>	3	3
<ul style="list-style-type: none"> <li>• Versioning and Version Control <ul style="list-style-type: none"> <li>○ need for versioning</li> <li>○ basic version control functions</li> </ul> </li> </ul>	3	4

Total: 39 hours

**School of Computing and Information Sciences**  
**COP 4814**  
**Component-Based Software Development**

**Course Outcomes Emphasized in Laboratory Projects / Assignments**

	<b>Outcome</b>	<b>Number of Weeks</b>
1	XML documents Outcomes: 1, 2	2
2	XSL translation Outcomes: 1, 2	2
3	Web Services Outcomes: 3, 6	6
4	Code access security Outcomes: 4	2
5	Versioning Outcomes: 5	1

**Oral and Written Communication:** No significant coverage

<b>Written Reports</b>		<b>Oral Presentations</b>
Approx. Number of pages	Number Required	Approx. Time for each
0	0	0

**Social and Ethical Implications of Computing Topics**

No significant coverage

<b>Topic</b>	<b>Class time</b>	<b>Student performance measures</b>

**School of Computing and Information Sciences**  
**COP 4814**  
**Component-Based Software Development**

**Estimate Curriculum Category Content (credit hours)**

<b>Fundamental IT Area</b>	<b>Core</b>	<b>Advanced</b>
Human computer interaction		
Information management		
Web systems and technologies	1.0	
System administration and maintenance		
Programming	1.0	
Networking		
Information assurance and security	0.5	
System integration and architecture	0.5	

**Theoretical Contents**

No Significant Coverage

**Problem Analysis Experiences**

1. Analyze application requirements and determine need for data exchange and transformation

**Solution Design Experiences**

1. Using XML/DOM API to integrate and exchange data
2. Using XSL/XSLT API to transform data
3. Implementation of applications that consume Web services