

1. Course number and name:

CMP 416 – Internet and Network Computing

2. Credits and contact hours

3 credit hours, 2 lab hour, 3 contact hours

3. Prerequisites and/or co-requisites

Prerequisites: CMP 310 (Introduction to Operating Systems) or COE 381 (Operating Systems), and CMP 320 (Database Systems) or COE 422(Database Systems), and COE 371 (Computer Networks I)

4. Name and Contact Information of Instructor:

Dr. Gerassimos Barlas

Office: EB2-210

Phone: Ext. 2558, 06-5152558

Email : gbarlas@aus.edu

Office Hours: Sun., Tue., Thu. 10:00-11:00 and 12:00-13:00

5. Course Description

Studies the design of Internet-based clients and servers; and multi-tiered applications, network application security; distributed object computing, remote method invocation, internet technology standards such as XML and Javascript; and building Internet-based applications.

6. Textbook and Other supplemental materials

Textbook:

Bryan Basham, Kathy Sierra and Bert Bates, “Head First Servlets and JSP”, O’Reilly Media, 2e, 2008, Packt Publishing ISBN13: 978-0596516680

Other supplemental materials

- Eric Jendrock, Ian Evans, Devika Gollapudi, Kim Haase and Chinmayee Srivathsa, *The Java EE 6 Tutorial: Basic Concepts*, 4e, Prentice Hall, 2010, ISBN-13: 978-0137081851
- Kathy Sierra and Bert Bates, *Head First EJB*, O’Reilly, 2003, ISBN-13: 978-0596005719
- David R. Heffelfinger, *Java EE 6 Development with NetBeans 7*, 1e, Packt Publishing 2011, ISBN-13: 978-1849512701
- Michael Mikowski, Josh Powell, *Single Page Web Applications: JavaScript end-to-end*, Manning Publications, 2013, ISBN-13: 978-161729075

7. Learning Outcomes

Upon completion of the course, students will be able to:

1. Design and implement server-side application logic using TCP/IP sockets, Remote Method Invocation (RMI), Java Server Faces (JSF) and Enterprise Java Beans (EJB).
2. Build Internet applications requiring Java DataBase Connectivity (JDBC) and Java Persistence API.
3. Implement the Model-View-Controller design pattern using JSF.
4. Construct applications that use XML technologies.

5. Add simple security mechanism to Web applications.
6. Use Javascript to programmatically control web clients.
7. Design and build Web applications using state-of-the-art development tools and web application frameworks.

8. Teaching and Learning Methodologies:

Methods include lectures, lab, problem based learning methods (homework/assignments), class discussions and a project.

9. Course Topics and Schedule:

Client/Server applications using TCP/IP sockets & remote objects	2 weeks
JDBC & Java Persistence API	1 week
HTTP Primer	1 week
XML	3 weeks
JSF	3 week
EJB	1.5 weeks
Security	1 week
Javascript/AJAX	2.5 week
Evaluation	1.5 weeks

10. Schedule of Laboratory and other Non-Lecture Sessions:

N/A

11. Out-of-Class Assignments with Due Dates:

Assignment 1: RMI	Week 4
Assignment 2: XML, XSLT & JSF	Week 6
Assignment 3: EJB	Week 9
Assignment 4: Javascript	Week 12
Project: Three-Tier (data server, application server, user interface) Web application	

12. Student Evaluation:

Student Evaluation:

Lab work	15%
Assignments	15%
Quizzes	15%
Midterm1 (March 27)	15%
Midterm2 (May 10)	15%
Final Project	25%

13. Contribution of Course to Program Outcomes

Program outcome

Emphasis in this course

- | | |
|---|----------------------------------|
| a) An ability to apply knowledge of computing and mathematics appropriate to the discipline | <input type="radio"/> |
| b) An ability to analyze a problem, and identify and define requirements | <input checked="" type="radio"/> |
| c) An ability to design, implement, and evaluate a computer-based system | <input checked="" type="radio"/> |

- d) An ability to function effectively on teams to accomplish a common goal
 - e) An understanding of professional, ethical, legal, security and social issues
 - f) An ability to communicate effectively with a range of audiences
 - g) An ability to analyze the local and global impact of computing
 - h) Recognition of the need for and an ability to engage in continuing professional level.
 - i) An ability to use current techniques, skills, and tools necessary for computing practice.
 - (j) An ability to apply mathematical foundations, algorithmic principles, and computer science
 - (k) An ability to apply design and development principles in the construction of software
- Emphasis: ● High; ● Medium; ○ Low; Blank – Nothing Specific Expected