

COURSE DESCRIPTION

Department and Course Number	CS420	Course Coordinator	Chengyu Sun
Course Title	Web Applications Architecture	Total Credits	4

Current Catalog Description:

Covers architectural considerations, design patterns, and implementation techniques in the development of enterprise web applications, as well as technologies that address productivity, stability, flexibility, scalability, and security.

Textbook:

Alur, Deepak and Crupi, John and Malks, Dan. *Core J2EE Patterns: Best Practices and Design Strategies*, Pearson Education, 2001.

References:

- Gamma, Erich and Helm, Richard and Johnson, Ralph and Vlissides, John. *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison Wesley Longman, 1998.
- Gehrtland, Justin and Tate, A. Bruce. *Better, Faster, Lighter Java*, O'Reilly, 2004.
- Johnson, Rod. *J2EE Design and Development*, Wrox, 2002.
- Johnson, Rod and Hoeller, Juergen and Arendsen, Alef and Risberg, Thomas and Sampaleanu, Colin. *Professional Java Development with the Spring Framework*, Wrox, 2005.
- Bauer, Christian and King, Gavin. *Hibernate in Action*, Manning Publications, 2004.
- Bill Burke, and Labourey, Sacha and Monson-Haefel, Richard. *Enterprise JavaBeans*, O'Reilly, 1999.

Course Goals:

At the end of the course, students are able to

- Understand the Model-View-Controller (MVC) architecture and common J2EE design patterns.
- Have in-depth knowledge of two to three mainstream web application frameworks such as Struts, Spring, and EJB.
- Understand alternative and/or emerging technologies such as Tapestry and JSF.

- Master build, logging, search, unit test, and other tools related to web development.

These course goals contribute to the success of **Student Learning Outcomes 2, 3, 5, and 6.**

Prerequisites by Topic:

- Strong Java programming skills
- Advanced knowledge of Java Servlet and JSP programming
- Basic knowledge of client-side web development
- Basic knowledge of relational databases and SQL

Major Topics Covered in the Course:

- MVC architecture
- J2EE design patterns
- Web application frameworks
 1. Jakarta Struts
 2. Spring
 3. Enterprise JavaBeans (EJB)
 4. Tapestry
 5. Java ServerFaces (JSF)
- Development tools and technologies
 1. Eclipse and Ant
 2. Commons-logging and log4j
 3. JUnit and JMeter
 4. Hibernate and JDO
 5. Lucene

Laboratory Projects (specify number of weeks on each):

Each week the students either work on a homework project or complete a 2-hour in-class lab on selected topics:

- Week 1: Familiarize with the server setup, application deployment, and the development environment.
- Week 2 and 3: Develop a web application using Struts. The application should include all tiers of a typical MVC application, which include business logic, data access, control flow, and presentation.

- Week 4 and 5: Add logging and validation support, and replace JDBC with Hibernate as the data access layer.
- Week 6 and 7: Replace Struts with Spring as the web application framework.
- Week 8 and 9: Re-write the application in EJB.
- Week 10: Add extra features or experiment with alternative technologies.

Estimate Curriculum Category Content (Quarter Hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms		0.5	Data Structures		0.5
Software Design		3.0	Prog. Languages		
Comp. Arch.					

Oral and Written Communications:

Written documentation of software built in labs and homework assignments.

Social and Ethical Issues:

No significant component.

Theoretical Content:

No significant component.

Problem Analysis:

Students are required to analyze the feature requirements of typical web applications, such as interactivity, work flow, and data management, and to identify the appropriate tools and technologies to implement these features.

Solution Design:

For a single application, students are required to implement the MVC architecture and design patterns using different web frameworks and combinations of tools. Furthermore, students are required to compare different implementations in terms of productivity, performance, flexibility, and scalability based on theoretical analysis and their own development experience.