

Architecting and Designing J2EE Applications (SL-425)

This course provides students with knowledge needed to use the Java 2 Platform, Enterprise Edition (J2EE platform) to create robust enterprise applications that allow for rapid change and growth. Participants gain an understanding of the strategies needed to create application blueprints that work well when implementing J2EE technologies. These strategies include effective decision making through the use of systemic qualities (such as scalability and flexibility), J2EE technology blueprints and design patterns, and iterative and incremental development.

Who Can Benefit

Students who can benefit from this course are those who are responsible for the overall software architecture and design of J2EE technology-based enterprise software systems. These students would also require insight into the role of the enterprise architect and want to use J2EE technologies in n-tier enterprise systems. Existing architects can also benefit from an understanding of how to use J2EE technologies to improve quality of service in their enterprise systems.

Prerequisites

To succeed fully in this course, students should be able to:

- Describe in outline form all J2EE technologies, including Enterprise JavaBeans, servlets, and JavaServer Pages
 - Describe distributed computing and communication concepts
 - Perform analysis and design of object-oriented software systems
 - Describe iterative and incremental software development
 - Use UML notation for modeling object-oriented systems
-

Skills Gained

Upon completion of this course, students should be able to:

- Make good use of J2EE component technologies to solve typical problems in system architecture
- Derive software systems using techniques outlined in the J2EE Blueprint and solutions defined in the J2EE Patterns Catalog
- Address quality-of-service requirements in a cost-effective manner using engineering trade-off techniques
- Describe the role of the architect and the products an architect delivers

- List and describe typical problems associated with large-scale enterprise systems
 - Identify the benefits derived from using an iterative and incremental development process
-

Related Courses

Before:

- [Developing Applications for the J2EE Platform \(FJ-310\)](#)
- [Advanced Business Component Development With Enterprise JavaBeans Technology \(SL-351\)](#)
- [J2EE Patterns \(SL-500\)](#)

After:

- [Developing Applications for the J2EE Platform \(FJ-310\)](#)
 - [Advanced Business Component Development With Enterprise JavaBeans Technology \(SL-351\)](#)
 - [J2EE Patterns \(SL-500\)](#)
-

Course Content

Module 1 - Introducing Fundamental Architectural Concepts

- Justify the need for architecture
- Describe the architectural function
- Describe five primary concerns of architecture

Module 2 - Examining System Architecture Development Heuristics and Guidelines

- Identify key risk factors in distributed enterprise systems
- Describe essentials of object-oriented design
- Describe guidelines for effective network communication
- Justify the use of transactions to control shared resources
- Describe controlling costs through concept reuse
- Explain the basic characteristics of Sun Open Net Environment (Sun ONE) architecture

Module 3 - Describing the Software Development Process

- Define the purpose of a development methodology
- Describe best practices of development methodologies
- Describe workflows and the four key phases of an iterative and incremental development process
- Define iterative process outputs
- Describe how to increase the probability of project success

Module 4 - Differentiating Local Object-Oriented Computing and Distributed Object Computing

- Examine the characteristics of data transfer in object-oriented computing
- Examine the characteristics of data transfer in distributed computing
- Compare and contrast distributed, object-based computing and object-oriented computing
- Analyze the impact of transactions in a distributed computing environment
- Analyze the impact of security implementation in a distributed computing environment

Module 5 - Optimizing Inside the Tier

- Plan system capacity to support required load
- Plan scalability to support anticipated growth
- Specify service providers to support maintenance goals

Module 6 - Optimizing Tier-to-Tier Communication

- Create a network model for an application
- Construct a data and call model to efficiently submit service requests and receive responses
- Create a transaction model for the application
- Create a security model for the application

Module 7 - Analyzing Quality-of-Service Requirements

- Evaluate the effects of dimensions on systemic qualities
- Prioritize quality-of-service requirements
- Inspect quality-of-service requirements for trade-off opportunities

Module 8 - Building a Prototype

- Create system prototypes that can potentially fulfill required architectural goals
- Assess the prototypes for fulfillment of required architectural goals

Module 9 - Describing J2EE Technology Servers

- Describe various server types
- Review enterprise information systems
- Explain the issues relating to the selection of application servers
- Describe deployment best practices, guidelines, and patterns

Module 10 - Describing J2EE Technologies

- Describe the J2EE technology web-tier components
- Describe the J2EE technology business-tier components
- Describe J2EE technology connectors

Module 11 - Evaluating the J2EE Technology Choice

- Design for long-term application state
- Manage client session state
- Enable business process and workflow control
- Enable presentation process and workflow control
- Manage presentation layout
- Design for asynchronous communication

Module 12 - Analyzing the Software Requirements

- Describe the software development process in terms of model transformations
- Gather requirements to define the business problem
- Create the domain model

Module 13 - Creating the Software Design

- Explain the purpose and elements of a robustness analysis and the design model
- Use robustness analysis to create a design model for a use case
- Generate a sequence diagram view of the design model

Module 14 - Creating the Software Architecture

- Describe the architecture model
- Decompose the system into components
- Arrange components into effective topologies
- Create deployment diagrams to document the architecture model
- Evaluate the architecture model

Module 15 - Choosing J2EE Technology Patterns

- Review relevant pattern fundamentals
- Recognize circumstances that suggest the use of J2EE technology patterns to assist in communications
- Recognize circumstances that suggest the use of J2EE technology patterns to assist in flexibility