

JB295 JBoss Enterprise Application Development Course Summary

Description

Intended for entry to mid-level Java™ developers, JBoss Enterprise Application Development (JB295) course will expose students to JBoss® Java EE frameworks, specifications, and interfaces (APIs). Students will learn how to create and maintain Java EE-compliant applications from start to finish using the Eclipse-based JBoss Developer Studio. Comprehensive lecture and extensive use-case, a hands-on lab introduces the students to Seam, Hibernate, and other related technologies that help create a fully functional enterprise Java application. By the end of the course, students will build a complete multi-tier enterprise application, including a web-based front end, a web services layer, EJB-layer, and a persistence layer, as well as code for test cases at all levels.

Topics

- Introduction to the JEE Application Stack, and the JBoss EAP server's technologies with a focus on Web UI development.
- Unit Testing tools and methodologies, as well as integration of tests with JBDS and Ant for continuous testing.
- Beginning of discussion of the business layer, including topics of code separation, business logic, and how the JEE stack separates the various layers.
- Details of JEE's asynchronous messaging technologies, compare and contrast various messaging constructs, and how to integrate messaging into an application.
- Representation of business data in Java, as well as managing transactions to the database layer. Final portion of the three-tier architecture, the persistence layer.
- How to expose business services as Web Services, for simpler distribution of the application functionality to outside businesses. Available tooling for the consumption of Web Services via JBoss is also covered.
- Using Seam as a migration tool for an application. This unit begins our discussion of JBoss enhancements to JEE development, to make the process more streamlined.
- Explore the various security features available in JBoss EAP 5.0.
- Advanced data model concepts are introduced, supported by Seam tooling, to produce robust web application features.
- A survey of integration testing technology and concepts.
- Examine how caching can enhance the functionality of a web application. Some of the pitfalls of caching are discussed, and JBoss tooling for caching, and how to use it, is covered.
- Advanced UI features are covered, including details on usability, and designer-provided interfaces.

Audience

- Students with existing Java programming experience seeking to enter Java Enterprise Edition (Java EE) development
- Experienced JEE developers who are migrating to JBoss for the first time

Prerequisites

- Fluency in HTML and Java programming language (Java SE, Java SE 5)
- Basic experience with an integrated development environment (IDE) such as Eclipse or NetBeans; and build tools such as Ant or Maven
- Basic knowledge asynchronous JavaScript (AJAX) and Relational Database Management System (RDBMS)

Duration

Four days

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

JB295 JBoss Enterprise Application Development Course Outline

I. Introduction to the JEE Application Stack, and the JBoss EAP server's technologies with a focus on Web UI development.

- A. JEE API
- B. JBoss EAP
- C. JSF
- D. Taglibs

II. Unit Testing tools and methodologies, as well as integration of tests with JBDS and Ant for continuous testing.

- A. JUnit/TestNG
- B. DBUnit
- C. Hibernate Tools
- D. JSFUnit
- E. Ant

III. Beginning of discussion of the business layer, including topics of code separation, business logic, and how the JEE stack separates the various layers.

- A. EJB3
- B. Session Beans
- C. JNDI
- D. JTA

IV. Details of JEE's asynchronous messaging technologies, compare and contrast various messaging constructs, and how to integrate messaging into an application.

- A. JMS
- B. JNDI
- C. Message Driven Beans

V. Representation of business data in Java, as well as managing transactions to the database layer. Final portion of the three-tier architecture, the persistence layer.

- A. Hibernate
- B. JPA
- C. Entity Beans
- D. JTA

VI. How to expose business services as Web Services, for simpler distribution of the application functionality to outside businesses. Available tooling for the consumption of Web Services via JBoss is also covered.

- A. Web Services

- B. JAX-WS
- C. SAAJ
- D. REST
- E. SOAP

VII. Using Seam as a migration tool for an application. This unit begins our discussion of JBoss enhancements to JEE development, to make the process more streamlined.

- A. Seam
- B. Seam-gen

VIII. Explore the various security features available in JBoss EAP 5.0.

- A. JAAS
- B. HTTP Authentication
- C. Seam

IX. Advanced data model concepts are introduced, supported by Seam tooling, to produce robust web application features.

- A. Data Model
- B. Annotations
- C. Bijection
- D. Seam Conversations

X. A survey of integration testing technology and concepts.

- A. SeamTest
- B. TestNG
- C. Ant

XI. Examine how caching can enhance the functionality of a web application. Some of the pitfalls of caching are discussed, and JBoss tooling for caching, and how to use it, is covered.

- A. JBoss Cache

XII. Advanced UI features are covered, including details on usability, and designer-provided interfaces.

- A. RichFaces
- B. Ajax4JSF
- C. Facelets