



Home > Training > Certification > Java Technology >

Sun Certified Enterprise Architect for the Java EE 5: Assignment (Step 2 of 3) (CX-310-301A)

Product Description

The Sun Certified Enterprise Architect for the Java Platform, EE 5 Technology (Step 2 of 3) certification exam is for enterprise architects responsible for architecting and designing Java Enterprise Edition (Java EE) platform compliant applications, which are scalable, flexible and highly secure.

Description of the Sun Certified Enterprise Architect Assignment Exam

The Sun Certified Enterprise Architect for Java EE (SCEA) is intended for Java technology professionals responsible for architecting systems. It is expected but not mandatory that a person taking the SCEA exam has already passed the Java technology programmer and Java technology developer certifications. The SCEA exam is broken down into three parts. Part 1 is multiple choice and tests your knowledge in the areas of general architecture and Java EE. Part 2 is an assignment that tests your ability to apply the knowledge you were tested on in part 1. Part 3 is an essay that asks questions about your assignment.

Description of the Sun Certified Enterprise Architect Assignment

This is a description of the nature, scale, and scope of the problem you will be required to solve in the SCEA assignment. These notes do not form any part of the actual assignment, and if you find a contradiction between these notes and the actual assignment instructions, you must adhere to the instructions in the assignment documents.

Nature of the Assignment

The assignment requires that you architect a solution for a small but plausible business system. To keep the amount of work involved to a reasonable level, the programs you create will be much more restricted in capability, and much cruder in overall presentation, than anything you would actually create for a paying customer. However, the essence of the problem will be the same. You will be graded on correctly solving the technical and performance requirements, not on the "polish" of the finished product.

Scale of the Assignment

Obviously, the amount of time taken by a candidate to create a solution to the assignment varies greatly. A fast candidate might create a solution in about 40 hours, but a more typical time requirement might be about 80 hours. If you spend a lot more than 100 hours on the project, then you might be creating something that is more detailed than the requirement, or you are lacking some of the skills necessary to complete the assignment.

Scoring Criteria

You will need to score 71% (114/160) to pass parts 2 and 3. Your project will be evaluated on a large number of objective criteria that fall into five sections:

- 1) Class Diagram: This section covers how well your class diagram(s) address the object model needed to satisfy the requirements.
- 2) Component Diagram: This section covers how well your component diagram(s) convey the structure of the architecture in satisfying the requirements.
- 3) Deployment Diagram: This section covers how well your deployment diagram conveys the production environment in satisfying the requirements.
- 4) Sequence/Collaboration Diagrams: This section covers how well your sequence or collaboration diagrams satisfy the requirements of the assignment.

5) Risk and Mitigation List: This section covers how well you have identified and mitigated risks that are involved in architecting the solution.

Additionally, categories 1-4 are evaluated on UML compliance.

APIs relevant to the Assignment

The assignment requires that you have an in-depth understanding of the Java programming language and JEE APIs. You can choose to use any released version of the JEE APIs.

Suggested Reading Materials

Mark Cade and Simon Roberts. Sun Certified Enterprise Architect for J2EE Technology Study Guide, Third Edition, Prentice Hall, 2002.

Erich Gamma. Design Patterns, Boston: Addison-Wesley Publishing Co., 1995.

Martin Fowler. UML Distilled: A Brief Guide to the Standard Object Modeling Language, Third Edition, Boston: Addison-Wesley Publishing Co., 2003.

Ramesh Nagappan, Robert Skoczylas, Rima Patel Sriganesh. Developing Java Web Services: Architecting and Developing Secure Web Services Using Java

<http://www.EnterpriseIntegrationPatterns.com/>

Nicholas Kassem, Enterprise Team. Designing Enterprise Applications with the Java 2 Platform, Enterprise Edition, First Edition (June 2, 2000), Addison-Wesley Pub Co

Enterprise Blueprints: <http://java.sun.com/blueprints/enterprise/>

» [Details below](#)

Assignments purchased on the Web site may only be used in the US. If you reside outside the US please [select a country](#) to inquire about products delivered in your country.

Details

- Delivered at: Delivered at: CertManager Database
- Prerequisites: Successful completion of the exam in Step 1 (CX-310-052)
- Other exams/assignments required for this certification: Other exams/assignments required for this certification: Step 1 (CX-310-052), Step 3 (CX-310-062)
- Exam type: Assignment
- Number of questions: n/a
- Pass score: 71%
- Time limit: 12 months from assignment download

Pass score is subject to the evaluation of the Step 3 essay exam and validation of the authenticity of the assignment.

Languages

English

Supporting Courses

No supporting courses currently available.

Assignment Objectives

Section 1: Application Design Concepts and Principles

- Document a given system architecture by creating UML diagrams for it
- Explain the main advantages of an object-oriented approach to system design. including the effect of encapsulation, inheritance, and use of interfaces on architectural characteristics.

- Describe how the principle of "separation of concerns" has been applied to the main system tiers of a Java Platform, Enterprise Edition (Java EE) application. Tiers include client (both GUI and web), web (web container), business (EJB container), integration, and resource tiers.
- Describe how the principle of "separation of concerns" has been applied to the layers of a Java EE application. Layers include application, virtual platform (component APIs), application infrastructure (containers), enterprise services (operating system and virtualization), compute and storage, and the networking infrastructure layers.

Section 2: Common Architectures

- Explain the advantages and disadvantages of two-tier architectures when examined under the following topics: scalability, maintainability, reliability, availability, extensibility, performance, manageability, and security.
- Explain the advantages and disadvantages of three-tier architectures when examined under the following topics: scalability, maintainability, reliability, availability, extensibility, performance, manageability, and security
- Explain the advantages and disadvantages of multi-tier architectures when examined under the following topics: scalability, maintainability, reliability, availability, extensibility, performance, manageability, and security.
- Explain the benefits and drawbacks of rich clients and browser-based clients as deployed in a typical Java EE application.
- Create a logical and physical model of a system infrastructure architecture.

Section 3: Integration and Messaging

- Explain possible approaches for communicating with an external system from a Java EE technology-based system given an outline description of those systems and describe the benefits and drawbacks of each approach.
- Explain typical uses of web services and XML over HTTP as mechanisms to integrate distinct software components.
- Explain how JCA and JMS are used to integrate distinct software components as part of an overall Java EE application.
- Given a scenario, explain the appropriate messaging strategy to satisfy the requirements

Section 4: Business Tier Technologies

- Explain and contrast uses for entity beans, entity classes, stateful and stateless session beans, and message-driven beans and understand the advantages and disadvantages of each type.
- Explain and contrast the following persistence strategies: container-managed persistence (CMP), BMP, JDO, JPA, and ORM, and using DAOs (Data Access Objects) and direct JDBC technology-based persistence under the following headings: ease of development, performance, scalability, extensibility and security.

Section 5: Web Tier Technologies

- Given a system requirements definition, explain and justify your rationale for choosing a web-centric or EJB-centric implementation to solve the requirements. Web-centric means that you are providing a solution that does not use EJBs. An EJB component-centric solution will require an application server that supports EJB components.

Section 6: Applicability of Java EE Technology

- Given a specified business problem, design a modular solution that solves the business problem using Java EE technology.
- Given a specified business problem, identify and prioritize the main technology risk areas that must be addressed by the technical design and architecture.
- Explain how the Java EE platform enables service-oriented architecture (SOA) -based applications.
- Identify how the Java SE and Java EE platforms support the internationalization and localization of applications.
- Explain your rationale for choosing build as compared to buy for a given Java EE component
- Explain the typical challenges associated with the design and implementation of large scale enterprise software

systems and how Java EE technology addresses those challenges.

- Explain how you would design a Java EE application to repeatedly measure critical non-functional requirements and outline a standard process with specific strategies to refactor that application to improve on the results of the measurements.

Section 7: Patterns

- From a list, select the most appropriate pattern for a given scenario. Patterns are limited to those documented in the book - Alur, Crupi and Malks (2003). Core J2EE Patterns: Best Practices and Design Strategies 2nd Edition and named using the names given in that book.
- From a list, select the most appropriate pattern for a given scenario. Patterns are limited to those documented in the book - Gamma, Erich; Richard Helm, Ralph Johnson, and John Vlissides (1995). Design Patterns: Elements of Reusable Object-Oriented Software and are named using the names given in that book.
- From a list, select the benefits and drawbacks of a pattern drawn from the book - Gamma, Erich; Richard Helm, Ralph Johnson, and John Vlissides (1995). Design Patterns: Elements of Reusable Object-Oriented Software.
- From a list, select the benefits and drawbacks of a specified Core J2EE pattern drawn from the book – Alur, Crupi and Malks (2003). Core J2EE Patterns: Best Practices and Design Strategies 2nd Edition.

Section 8: Security

- Given an architectural system specification, select the appropriate locations for implementation of specified security features, and select suitable technologies for implementation of those features
- Identify and classify potential threats to a system and describe how a given architecture will address the threats.

Copyright 1994-2007 Sun Microsystems, Inc.