Certified Senior System Architect

EXAM BLUEPRINT

Pega® ACADEMY

This document is the property of:
Pegasystems Inc.
1 Rogers Street
Cambridge, MA 02142
Phone: (617) 374-9600
Fax: (617) 374-9620
www.pega.com

Document: Certified Senior System Architect (7.1)
Updated: January 19, 2015

The Pegasystems Certified Professional Program

The Pegasystems Certified Professional Program has created a worldwide community of thousands of Certified Professionals, drawn by a commitment to excel in their ability to deliver world class Pega applications.

When a Senior System Architect participates in the design and development of a Pega application there are fundamental, essential Pega skills and knowledge areas that must be applied to ensure success. It is these skills and knowledge areas that form the basis of the certification exam.

Thus, Pegasystems is committed to providing you the training, tools, and knowledge that you need to achieve Certification as a Senior System Architect.

Path to the Senior System Architect Certification

About the Exam Blueprint

The purpose of the blueprint is to provide you with a roadmap of the Senior System Architect Certification exam content to allow you to better prepare for the exam.

The blueprint includes test domain weighting, test objectives, and topical content. The topics and concepts are included to clarify the test objectives.

The exam is based upon the knowledge areas necessary for a Senior System Architect to be able to participate successfully design and development of Pega applications.

Candidates are tested on their:

- Ability to analyse requirements to design and create application components
- Ability to apply key concepts and techniques in the design and construction of the components of a multi-process application
- Ability to understand and apply Pega application design and case design principles and best practices on projects
- Ability to address security/access requirements in an application
- Understanding of application debugging tools and features
- Understanding of performance tuning and system maintenance techniques and tools.

Prerequisites

The suggested training prerequisites for this certification level are:

- Certified System Architect
- CSSA eLearning or Immersion (Instructor-Led)

The skills and knowledge areas measured by this exam are derived directly from the content of these courses.

Exam Test Domains

The table below lists the test domains and the extent to which they are represented as an estimated percentage of test items.

Test Domains	% of Exam
Application Design	11.4%
Case Design	15.7%
Data Model	10%
Automating Business Policies	20%
User Experience	12.9%
User Reporting	7.1%
Integration	8.6%
Architecture	5.7%
Administration	8.6%
Total	100%

Exam Format

The exam consists of 73 multiple choice questions with three questions that are experimental items and are not scored. You are given 90 minutes to complete the exam and the Non-Disclosure Agreement (NDA) requirement. A passing score of 70% is needed to be recognized as a Pega Certified Senior System Architect.

Question Format

The examinee selects from one or more response options to answer a question. A response is considered correct when it accurately completes the statement or answers the question. Distracters or incorrect answers are plausible response options that examinees with incomplete knowledge are likely to choose.

Test item formats used in this examination are:

- **Multiple Choice** — Select one option that best answers the question or completes a statement.
- **Multiple Responses** — Select more than one option that best answers the question or completes a statement. The text states how many options are correct, such as Choose two.
- **Sample Directions** — Read the statement or question. From the response options, select the option(s) that represent the most correct or best answer(s) given the information provided.
- **True/False —** Read the statement or question. Select either true or false as the answer.

Test Topics

Application Design

- Application Express
- Rulesets
  - Ruleset Versioning
  - Branch Rulesets
- Rule Resolution
  - Rule Availability
- Enterprise Class Structure
  - Enterprise Class Structure Best Practices
  - Enterprise Class Structure Layers (Framework, Implementation)
- Reusability and Specialization
  - Importance of Rule Specialization and Reuse
  - Rulesets and Rule Specialization

Case Design

- Case management
- Case Lifecycle management
  - Case Stage configurations
  - Configuring steps in each case stage
- Case Hierarchy
  - Case and subcase relationships
  - Case Instantiation
  - Data Propagation
  - Locking
- Process Flow Modelling
  - Flow Shapes
  - Screen Flows
  - Flow Configuration
  - Work Status
  - Work Parties

Data Model

- Property Definitions
  - Property Types
  - Property Modes
  - Auto-Populate
- Data Pages
  - Data Page Sourcing
  - Data Page Refresh strategy
  - Date Page structure
  - Referencing Data Page
- Data Modelling

Automating Business Policies

- Declarative Rules
- Declarative Expressions
  - Forward Chaining
  - Backward Chaining
- Decisioning Rules
  - Decision Tables
  - Decision Trees
  - Map Values
  - When Rules
- Delegating Decision rules to business users
- Case Processing Rules
  - Data Transforms
  - Activities
- Advanced Case Processing
  - Routing
  - PegaPulse
  - Case Attachments
  - Ticketing
  - Validation
  - Correspondence

User Experience

- UI Theory
  - UI Best Practices
  - UI Design Decisions
- UI Fundamentals
  - Harnesses, flow actions and sections
  - Dynamic Layouts
  - Grids
- Formatting UI using Controls
- Building Responsive UI
- Implementing Dynamic Actions
  - Action sets
  - Event Architecture
  - Client-side dynamic UI

Reporting

- Reporting Basics
  - Report definitions, Report viewer
  - Filtering
  - Exposing Data
  - SQL functions
- Business User Reporting
- Reporting Data Model
- Charts

Integration

- Connectors
  - Processing Model
  - Simulation
- Services
  - Processing Model
  - Service Wizard
  - Unit testing
- Listeners
  - File listeners and services
  - File parsing
- Error Handling
- External Database Integration
  - When to use a Connect SQL
  - External class mappings

Architecture

- Agents
- Security
  - Access role
  - Application
  - Privilege
- LDAP Authentication

Administration

- System Debugging tools
  - Clipboard
  - Tracer
  - UI Inspector
  - System Management Application
- Performance Analysis Tools
  - PAL (Performance Analyzer)
  - Log Usage
  - Guardrail Reports
- System Logging
  - Accessing and reading logs
- Purge and Archive wizard
- Application Migration
  - Application Packaging wizard
  - Rule-Admin-Product
  - Export Gadget
  - Import wizard