Developing Applications using Cisco Core Platforms and APIs v1.0 (350-901)

Exam Description: Developing Applications using Cisco Core Platforms and APIs v1.0 (DEVCOR 350-901) is a 120-minute exam associated with the DevNet Professional Certification. This exam tests a candidate's knowledge of software development and design including using APIs, Cisco platforms, application deployment and security, and infrastructure and automation. The course, Developing Applications using Cisco Core Platforms and APIs helps candidates to prepare for this exam.

The following topics are general guidelines for the content likely to be included on the exam. However, other related topics may also appear on any specific delivery of the exam. To better reflect the contents of the exam and for clarity purposes, the guidelines below may change at any time without notice.

20% 1.0 Software Development and Design

- 1.1 Describe distributed applications related to the concepts of front-end, back-end, and load balancing
- 1.2 Evaluate an application design considering scalability and modularity
- 1.3 Evaluate an application design considering high-availability and resiliency (including on-premises, hybrid, and cloud)
- 1.4 Evaluate an application design considering latency and rate limiting
- 1.5 Evaluate an application design and implementation considering maintainability
- 1.6 Evaluate an application design and implementation considering observability
- 1.7 Diagnose problems with an application given logs related to an event
- 1.8 Evaluate choice of database types with respect to application requirements (such as relational, document, graph, columnar, and Time Series)
- 1.9 Explain architectural patterns (monolithic, services oriented, microservices, and event driven)
- 1.10 Utilize advanced version control operations with Git
  - 1.10.a Merge a branch
  - 1.10.b Resolve conflicts
  - 1.10.c git reset
  - 1.10.d git checkout
  - 1.10.e git revert
- 1.11 Explain the concepts of release packaging and dependency management
- 1.12 Construct a sequence diagram that includes API calls

20% 2.0 Using APIs

- 2.1 Implement robust REST API error handling for time outs and rate limits
- 2.2 Implement control flow of consumer code for unrecoverable REST API errors
- 2.3 Identify ways to optimize API usage through HTTP cache controls
- 2.4 Construct an application that consumes a REST API that supports pagination
- 2.5 Describe the steps in the OAuth2 three-legged authorization code grant flow

20% 3.0 Cisco Platforms

- 3.1 Construct API requests to implement chatops with Webex Teams API
- 3.2 Construct API requests to create and delete objects using Firepower device management (FDM)
- 3.3 Construct API requests using the Meraki platform to accomplish these tasks
  - 3.3.a Use Meraki Dashboard APIs to enable an SSID
  - 3.3.b Use Meraki location APIs to retrieve location data
- 3.4 Construct API calls to retrieve data from Intersight
- 3.5 Construct a Python script using the UCS APIs to provision a new UCS server given a template
- 3.6 Construct a Python script using the Cisco DNA center APIs to retrieve and display wireless health information
- 3.7 Describe the capabilities of AppDynamics when instrumenting an application
- 3.8 Describe steps to build a custom dashboard to present data collected from Cisco APIs

20% 4.0 Application Deployment and Security

- 4.1 Diagnose a CI/CD pipeline failure (such as missing dependency, incompatible versions of components, and failed tests)
- 4.2 Integrate an application into a prebuilt CD environment leveraging Docker and Kubernetes
- 4.3 Describe the benefits of continuous testing and static code analysis in a CI pipeline
- 4.4 Utilize Docker to containerize an application
- 4.5 Describe the tenets of the "12-factor app"
- 4.6 Describe an effective logging strategy for an application
- 4.7 Explain data privacy concerns related to storage and transmission of data
- 4.8 Identify the secret storage approach relevant to a given scenario
- 4.9 Configure application specific SSL certificates
- 4.10 Implement mitigation strategies for OWASP threats (such as XSS, CSRF, and SQL injection)
- 4.11 Describe how end-to-end encryption principles apply to APIs

20% 5.0 Infrastructure and Automation

- 5.1 Explain considerations of model-driven telemetry (including data consumption and data storage)
- 5.2 Utilize RESTCONF to configure a network device including interfaces, static routes, and VLANs (IOS XE only)
- 5.3 Construct a workflow to configure network parameters with:
  - 5.3.a Ansible playbook
  - 5.3.b Puppet manifest
- 5.4 Identify a configuration management solution to achieve technical and business requirements
- 5.5 Describe how to host an application on a network device (including Catalyst 9000 and Cisco IOx-enabled devices)