



CERTIFICATION SYLLABUS

Certified API Automation Specialist

Exam Code: CST-API™



A complete preparation guide covering exam objectives, core competencies, domain weightage, and topics to prepare.

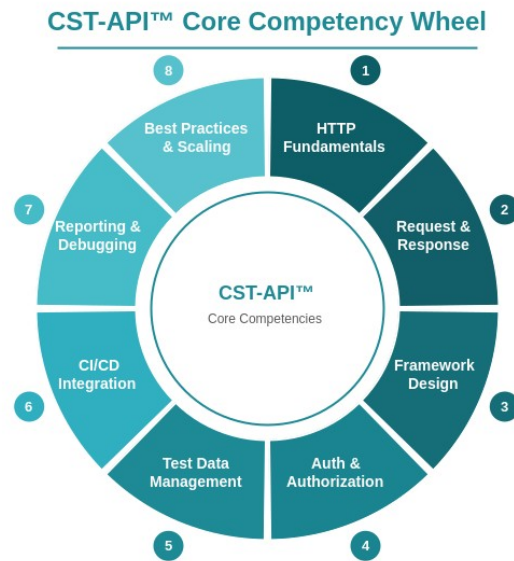
Table of Contents

1. Exam Name & Code.....	1
2. Table of Contents.....	2
3. Objective of the Certification.....	3
4. Core Competencies.....	4-6
➤ API Fundamentals & HTTP Protocol	
➤ Request Construction & Response Validation	
➤ API Test Framework Design	
➤ Authentication & Authorization Testing	
➤ Test Data Management & Data-Driven API Testing	
➤ CI/CD Integration & DevOps Practices	
➤ Reporting, Logging & Debugging	
➤ Best Practices, Maintenance & Scalability	
5. Who Should Take This Certification.....	7
6. Exam Information.....	7
7. Examination Domain & Weightage.....	8
8. Topics to Prepare.....	9-10
➤ API Fundamentals & HTTP Protocol	
➤ Request Construction & Response Validation	
➤ API Test Framework Design	
➤ Authentication & Authorization Testing	
➤ Test Data Management & Data-Driven Testing	
➤ CI/CD Integration & DevOps Practices	
➤ Reporting, Logging & Debugging	
➤ Best Practices, Maintenance & Scalability	

Objective of the Certification

- **Build a strong foundation in API fundamentals and the HTTP protocol** — understanding request methods, status codes, headers, and how RESTful/SOAP APIs communicate.
- **Develop proficiency in request construction and response validation** — sending requests and validating responses (status codes, headers, body content) using assertions.
- **Establish competency in API test framework design** — building maintainable, scalable automation frameworks using tools such as REST Assured, Postman/Newman, or similar.
- **Build skills in authentication and authorization testing** — validating APIs secured with API keys, Basic Auth, OAuth 2.0, JWT, and other authentication schemes.
- **Develop competency in test data management and data-driven API testing** — supplying and validating requests using external data sources and parameterization.
- **Introduce CI/CD integration practices** — integrating automated API test suites into continuous integration pipelines (e.g., Jenkins, GitHub Actions, Azure DevOps).
- **Build competency in reporting, logging, and debugging** — generating actionable test reports and diagnosing API test failures efficiently.
- **Establish best practices for framework maintainability and scalability** — writing reusable, robust code that scales with growing API surface area and test coverage.
- **Prepare candidates to contribute effectively to automation teams** — equipping them with practical skills to build, extend, and maintain real-world API test suites.
- **Create a foundation for progression toward advanced automation and SDET roles** — such as automation architecture, performance/load testing, or automation leadership positions.

Core Competencies — CST-API™ (API Automation Specialist)



1. API Fundamentals & HTTP Protocol

- Explain core HTTP concepts: methods (GET, POST, PUT, PATCH, DELETE), status codes, headers, and request/response structure.
- Distinguish between REST, SOAP, and GraphQL API styles at a conceptual level.
- Understand API documentation formats (e.g., Swagger/OpenAPI) and how to interpret them for test design.

2. Request Construction & Response Validation

- Construct API requests with appropriate headers, query parameters, path parameters, and request bodies.
- Validate response status codes, headers, and body content (JSON/XML) using assertions.
- Apply schema validation to ensure response structure conforms to expected contracts.

3. API Test Framework Design

- Design and implement API automation frameworks using tools such as REST Assured (Java) or Postman/Newman.
- Structure a framework (folder organization, configuration management, reusable request builders/utilities).
- Apply design principles that reduce duplication and improve framework scalability.

4. Authentication & Authorization Testing

- Test APIs secured with API keys, Basic Authentication, OAuth 2.0, and JWT-based authentication.
- Validate authorization rules (role-based access, permission scopes) and expected failure responses (401/403).
- Manage and refresh authentication tokens within automated test flows.

5. Test Data Management & Data-Driven API Testing

- Supply test data via external sources (JSON, CSV, Excel, databases) for parameterized API tests.
- Manage test environment configuration (base URLs, credentials, environment variables) across environments.
- Design tests that validate both positive and negative/edge-case data scenarios.

6. CI/CD Integration & DevOps Practices

- Integrate API test suites into CI/CD pipelines (e.g., Jenkins, GitHub Actions, Azure DevOps, GitLab CI).
- Configure automated test triggers (on commit, scheduled runs, pre-deployment gates).

- Understand how API test automation fits into a broader DevOps/continuous testing strategy.

7. Reporting, Logging & Debugging

- Generate and interpret test execution reports (e.g., Extent Reports, Allure, Newman HTML reports).
- Implement logging strategies to capture request/response details for troubleshooting.
- Debug common API test failures: authentication errors, schema mismatches, timeout/connectivity issues.

8. Best Practices, Maintenance & Scalability

- Apply coding best practices for readability, reusability, and maintainability of API automation code.
- Manage test data and environment configuration to support scalable execution across environments.
- Plan for framework evolution as APIs version, expand, or change over time.

Who Should Take This Certification

- Manual QA testers transitioning into API test automation roles.
- SDETs (Software Development Engineers in Test) building or maintaining API automation frameworks.
- QA Automation Engineers responsible for API/backend test suites.
- Backend developers contributing to test automation as part of a DevOps or Agile team.
- Automation leads/architects evaluating or standardizing team API testing practices.
- Individuals who have completed CSTF™ (or hold equivalent foundational testing knowledge) and want to specialize in API automation.

Exam Information

Exam Name	API Automation Specialist
Exam Code	CST-API™
Exam Format	Multiple Choice Questions (MCQ) & Code-Based/Practical Scenario Questions
Duration	120 minutes
Number of Questions	75
Passing Score	70%
Mode of Exam	Test Center Based Testing / Online Testing
Language	English
Validity	Lifetime
Open Book	No

Examination Domain & Weightage

Domain	Weightage
API Fundamentals & HTTP Protocol	15%
Request Construction & Response Validation	15%
API Test Framework Design	20%
Authentication & Authorization Testing	15%
Test Data Management & Data-Driven Testing	10%
CI/CD Integration & DevOps Practices	10%
Reporting, Logging & Debugging	8%
Best Practices, Maintenance & Scalability	7%
Total	100%

Topics to Prepare

The topics below represent the core areas candidates should focus on for each domain, reflecting the practical, hands-on depth expected at the CST-API™ level.

1. API Fundamentals & HTTP Protocol

- **HTTP Methods:** GET, POST, PUT, PATCH, DELETE and their appropriate use cases.
- **Status Codes:** Common 2xx, 3xx, 4xx, and 5xx codes and what they signify.
- **Headers & Structure:** Request/response headers, content types, and body structure.
- **API Styles:** REST vs. SOAP vs. GraphQL at a conceptual level.
- **API Documentation:** Reading and interpreting Swagger/OpenAPI specifications.

2. Request Construction & Response Validation

- **Building Requests:** Query parameters, path parameters, headers, and request bodies.
- **Response Assertions:** Validating status codes, headers, and JSON/XML body content.
- **Schema Validation:** Ensuring responses conform to expected data contracts/schemas.

3. API Test Framework Design

- **REST Assured Basics:** Structuring API test classes and reusable request specifications (Java).
- **Postman/Newman:** Building collections, environments, and running them via command line.
- **Framework Structure:** Organizing folders, configuration files, and reusable utility/request-builder classes.
- **Design Patterns:** Reducing duplication through builder patterns and reusable request specifications.

4. Authentication & Authorization Testing

- **Auth Schemes:** API keys, Basic Auth, OAuth 2.0, and JWT-based authentication.
- **Token Management:** Acquiring, refreshing, and reusing tokens across automated test flows.
- **Authorization Validation:** Testing role-based access and expected 401/403 failure responses.

5. Test Data Management & Data-Driven Testing

- **External Data Sources:** Supplying test data via JSON, CSV, Excel, or databases.
- **Environment Configuration:** Managing base URLs, credentials, and variables across environments.

- **Edge Case Design:** Structuring tests for both valid and invalid/edge-case input data.

6. CI/CD Integration & DevOps Practices

- **Pipeline Integration:** Incorporating API test suites into Jenkins, GitHub Actions, Azure DevOps, or GitLab CI.
- **Automated Triggers:** Running tests on commit, on a schedule, or as a pre-deployment gate.
- **Continuous Testing Mindset:** Understanding API automation's role within a broader DevOps strategy.

7. Reporting, Logging & Debugging

- **Test Reporting:** Generating and interpreting reports (Extent Reports, Allure, Newman HTML reports).
- **Logging Practices:** Capturing request/response details for diagnostic purposes.
- **Debugging Failures:** Diagnosing authentication errors, schema mismatches, and timeout/connectivity issues.

8. Best Practices, Maintenance & Scalability

- **Code Quality:** Writing readable, reusable, and maintainable API automation code.
- **Test Data & Environment Management:** Supporting execution across multiple environments.
- **Framework Evolution:** Planning for growth as APIs version, expand, or change over time.