



# Learn API Testing

Norms, Practices, and Guidelines for  
Building Effective Test Automation

Jagdeep Jain

Apress®

# **Learn API Testing**

**Norms, Practices, and  
Guidelines for Building  
Effective Test Automation**

**Jagdeep Jain**

**Apress®**

## *Learn API Testing: Norms, Practices, and Guidelines for Building Effective Test Automation*

Jagdeep Jain

Dewas, Madhya Pradesh, India

ISBN-13 (pbk): 978-1-4842-8141-3

ISBN-13 (electronic): 978-1-4842-8142-0

<https://doi.org/10.1007/978-1-4842-8142-0>

Copyright © 2022 by Jagdeep Jain

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr

Acquisitions Editor: Divya Modi

Development Editor: James Markham

Coordinating Editor: Divya Modi

Copy Editor: Mary Behr

Cover designed by eStudioCalamar

Cover image designed by Freepik ([www.freepik.com](http://www.freepik.com))

Distributed to the book trade worldwide by Springer Science+Business Media New York, 1 New York Plaza, Suite 4600, New York, NY 10004-1562, USA. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit [www.springeronline.com](http://www.springeronline.com). Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail [booktranslations@springernature.com](mailto:booktranslations@springernature.com); for reprint, paperback, or audio rights, please e-mail [bookpermissions@springernature.com](mailto:bookpermissions@springernature.com).

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at [www.apress.com/bulk-sales](http://www.apress.com/bulk-sales).

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at <https://github.com/Apress/Learn-API-Testing>.

For more detailed information, please visit <https://github.com/Apress/Learn-API-Testing>.

Printed on acid-free paper

*I dedicate this book to my teachers, mentors,  
and colleagues who have been instrumental in  
the enhancement of my knowledge on the subject,  
and also to my wife, daughter, sisters, parents, and in-laws,  
without whose relentless support it would not have been  
possible to manage the tight schedule of this work.*

*—Jagdeep Jain*

# Table of Contents

- About the Author .....xv**
- About the Technical Reviewers .....xvii**
- Acknowledgments .....xix**
- Introduction .....xxi**
  
- Chapter 1: Introduction to API Testing..... 1**
  - What Is API Testing?..... 1
  - Need.....5
  - Types of API Testing .....6
  - Advantages .....8
  - Summary.....9
  
- Chapter 2: Web Application Architecture ..... 11**
  - Web Applications Defined ..... 11
  - Monolithic vs. Microservices Architecture ..... 12
    - Designing Test Strategies ..... 17
  - RESTful Architecture ..... 18
  - HTTP.....20
  - Headers.....23
  - Requests .....25
    - Request Methods.....25
    - Resource Addresses .....25

## TABLE OF CONTENTS

Request Headers .....	25
Request Body .....	26
Response .....	27
Status Line.....	27
Response Header .....	28
Response Body .....	28
Response Codes.....	28
Summary.....	30
<b>Chapter 3: Authentication.....</b>	<b>31</b>
HTTP Authentication.....	31
Basic Authentication.....	32
Session-Based Authentication .....	32
Token/JWT-Based Authentication .....	33
OAuth2-Based Authentication .....	35
Authorization.....	37
RBAC.....	37
ABAC.....	38
Authentication and Authorization Services .....	38
Summary.....	39
<b>Chapter 4: Tools, Frameworks, and Libraries .....</b>	<b>41</b>
API Testing Tools .....	42
cURL .....	42
Postman .....	51
RestAssured .....	63
Frameworks/Libraries.....	71
TestNG .....	71
Log4j.....	72

Jackson-Databind .....	72
HashMap .....	72
Assertj .....	73
Java Spring.....	73
Summary.....	73
<b>Chapter 5: Test Pyramid .....</b>	<b>75</b>
Black Box Testing.....	76
Grey Box Testing.....	77
White Box Testing.....	77
Test Pyramid .....	78
Summary.....	79
<b>Chapter 6: Testing the API .....</b>	<b>81</b>
Workflows/Use Cases/Test Script .....	82
Schema Validation.....	82
Test Coverage.....	85
Header Testing .....	86
Request Header .....	86
Response Header .....	87
Request Body .....	88
Format Unsupported.....	88
Special Characters .....	88
Very Long Strings .....	88
Invalid Method .....	89
Invalid Value .....	89
Incorrect Data Type.....	89
Empty Data/Object.....	89
Required Fields.....	89

TABLE OF CONTENTS

- Null .....90
- Redundant Fields.....90
- DELETE Already Deleted Entity .....90
- Duplicate Check.....90
- Response Body .....91
  - Actual Data vs. Expected Data.....91
  - Limit/Size/Pagination/Sorting.....91
  - API Version Testing .....91
- Internal vs. External APIs .....91
- Consumer-Driven Contract Testing .....92
- Importance of Negative Testing .....92
- Summary.....93
- Chapter 7: A Good Test Script.....95**
  - Components of a Test Script .....96
    - setup( ) .....97
    - test( ) .....97
    - teardown( ) .....97
  - Guidelines .....98
    - Single-Attempt Test .....98
    - Document Test Objective .....98
    - Keep It Small .....99
    - Use assertj for Assertions.....99
    - Use log4j.....100
    - Order of Tests .....100
    - No Interventions Between Test Steps.....101
    - Avoid Hard Sleeps .....102
    - Always Use Assertions.....102



Do Not Overtest .....	103
Do Not Import a Test into Another Test .....	103
Test Boundaries .....	103
API Test Coverage .....	104
Provide Short Commands .....	104
Do not try{} catch{} .....	104
Summary.....	104
<b>Chapter 8: Coding Guidelines .....</b>	<b>105</b>
Coding Best Practices .....	105
Class Naming Conventions .....	106
Method Naming Conventions .....	106
Variable Naming Conventions.....	107
Constant Naming Conventions .....	107
Provide User Actions.....	107
Simplicity.....	107
Indentation .....	108
Test Assertions.....	108
Test Class Naming Conventions.....	109
Test Method Naming Conventions .....	109
Test Package Naming Conventions.....	111
Documentation.....	113
Summary.....	113
<b>Chapter 9: Organize a Test Framework .....</b>	<b>115</b>
Framework Requirements.....	116
Request .....	118
Response.....	118
Exception.....	118

## TABLE OF CONTENTS

Configuration .....	119
User Authentication .....	119
Processor.....	119
Model.....	120
Test Framework.....	120
Test Assertions .....	120
Logger .....	121
Util .....	121
Test Execution .....	121
Debug Config .....	121
Test Driver .....	122
Setting Up a Maven Project.....	122
Dependencies and Plugins.....	123
RestAssured .....	123
Log4j.....	123
TestNG .....	124
Spring Framework.....	124
Assertj .....	125
Jackson-Databind .....	125
Maven Compiler Plugin.....	125
Surefire Plugin.....	126
Java Code Formatting Plugin.....	127
Request.....	128
Response .....	131
Exceptions.....	135
Configuration.....	136
Properties File .....	136
Spring .....	137

Application Configuration .....	139
Application Context.....	140
Application Config .....	140
Complete URL For the Test Script.....	141
Test Data.....	142
User Authentication.....	144
Processor .....	145
Model .....	146
Test Framework .....	146
Logger.....	148
Util.....	149
Test Execution.....	149
Debug Config .....	150
Test Driver.....	150
Summary.....	150
<b>Chapter 10: First Test Script.....</b>	<b>151</b>
Developing Your First Test.....	151
Base Test .....	152
First Test.....	154
Test Suite .....	157
TestNG XML .....	157
Executing a Test.....	159
Execute a Test Suite .....	159
Execute an Individual Test.....	159
Execution Results.....	160
TestNG Report.....	160

## TABLE OF CONTENTS

Logging .....	161
log().all() .....	161
Response Time.....	162
Debug.....	163
Summary.....	164
<b>Chapter 11: API Documentation.....</b>	<b>165</b>
Need.....	166
Swagger.....	167
Summary.....	176
<b>Chapter 12: Case Study: Shopping Cart APIs.....</b>	<b>177</b>
Feature List .....	178
QA Responsibility Matrix .....	179
Sprint # .....	181
Goal Setting .....	183
Sprint One .....	183
Sprint Guidelines .....	184
QA Tasks .....	185
Targeted Features.....	189
API Endpoints .....	190
Unit Testing.....	190
Test Plan Development .....	190
Test Data Preparation .....	191
Manual Test Scripts .....	192
Postman .....	192
Test Automation.....	192
Test Suite.....	194
Parallel Test Execution.....	195

Test Execution ..... 196

Front-End Team ..... 197

Sprint Nth..... 197

    Sprint Demo Feedback Testing ..... 198

Hardening Sprint..... 198

    Release Testing..... 198

Summary..... 199

**Appendix A: Workstation Setup .....201**

    Java..... 201

        MacOS ..... 202

        Ubuntu ..... 202

        Linux..... 202

        Windows..... 202

    Maven ..... 203

        MacOS ..... 203

        Ubuntu ..... 204

        Linux..... 204

        Windows..... 204

    Maven Project..... 204

    cURL..... 205

        MacOS ..... 205

        Ubuntu ..... 205

        Linux..... 206

        Windows..... 206

    Postman ..... 206

    IDE..... 206

    Tomcat ..... 206

TABLE OF CONTENTS

    MacOS/Ubuntu/Linux.....206  
    Windows.....207  
**Appendix B: Contact Management Application .....209**  
    Swagger .....211  
**Appendix C: Shopping Cart Application .....213**  
    Swagger .....214  
**Index.....215**

# About the Author



**Jagdeep Jain** has a Bachelor of Computer Science and Engineering degree and more than 15 years of rich experience in the software quality assurance and testing domain. He has worked for several product development software companies. He is a firm believer and an avid advocate of test automation. He is also the co-author of *Pro Apache JMeter* with Sai Matam.

# About the Technical Reviewers



**Nitesh Kumar Jain** has over a decade of experience in the software testing world. He has an M.Tech in Information Technology from IIITM Gwalior, M.P. and a B.E. in Computer Science and Engineering from NIT Raipur, Chattisgarh. He is a keen technology learner with a “let’s automate everything” attitude. He is also an ISTQB-certified Test Manager, Technical Test Analyst, and Agile Test Engineer. He loves to make Java/Swing-based tools that can help with anything related to software testing. He is

presently working as a Lead QA at <https://watermarkinsights.com> and is constantly involved in doing quality work on framework design for UI, API, and performance test automation. His LinkedIn profile is <https://www.linkedin.com/in/nitesh-jain-958a2630/>.



## ABOUT THE TECHNICAL REVIEWERS



**Kushagra Mittal** has a Bachelor of Technology in Computer Science degree from Amity University, Lucknow, UP. He has over nine years of experience in developing back-end solutions for multinational companies and has built products that are used by thousands of customers. He has developed microservices using Java Spring Boot and has used the Swagger UI for API documentation. He has hosted various training sessions with the University of Lucknow, UP on big data, distributed systems, and machine learning. He is an Oracle-certified Java Programmer, Oracle

Cloud Infrastructure Foundation Certified Associate, and AWS Certified Developer - Associate. He is currently working as a Principal Member of Technical Staff at Oracle India Pvt. Ltd. His LinkedIn profile is <https://www.linkedin.com/in/kushagra-mittal/>.

# Acknowledgments

I want to thank everyone who helped in giving shape to this book, including but not limited to providing useful and timely feedback on the chapters, source code, and test scripts, and finding bugs in the sample web applications. Without them, it would have been tough to create *good quality work* for you, the reader.

Thanks to Nidhi Jain for reviewing each and every line of the book with *the* objective of improving readability.

Nitin Dhawan works as a program manager and he has helped various teams in setting up scrum best practices and implementing planning, monitoring, and risk assessment modules. Currently he is working as a technical program manager and is responsible for establishing the communication channel between engineering teams by ensuring regular communication on projects/programs and status to everyone. Thanks to Nitin for reviewing the case study chapter. It was a big help to get all of the angles on how the software industry works in the scrum model.

Thanks to Aashita Priya, Advocate Jayant R. Vipat, Akshay Muramatti, Amudhan Kash, Anand Sinha, Anuj Yadav, Arun Vijapur, Arijit Hawlader, Aruna Piraviperumal, Ashish Mankar, Beejal Vibhakar, Bharat Vipat, Deepika Sharma, Ganesh Phirke, Ganesh Prasath S, Gomtesh Gandhi, Gyan Bhal, Haridev Vengateri, Harshad Savot, Harshvardhan Vipat, Jay Erb, Jay Shah, Jaya Gopal Somu, Jon Gunnip, Kevendra Patidar, Laura King, Mangesh Lunawat, Marque Davis, Matt Armstrong, Michael Laube, Monica Poddar, Mukesh Bafna, Nehal Gaikwad, Nikhil Agrawal, Nitish Shirsath, Niti Dugar, Pankaj Saraf, Patrick Lee, Peeyush Janoria,

## ACKNOWLEDGMENTS

Piyush Singh, Prasad Jakka, Prasoon Kumar, Qian Li, Rajat Jain, Rangith Vaddepally, Ramanuj Vipat, Ramesh Sunkara, Rohit Bagde, Sai Matam, Sathya Gowri N, Shally Garg, Sharon Annese, Shravan Belde, Snehal Mundle, Stella Yun, Sudeep Tripathy, Tapan Upadhyay, Tarak Joshi, Tina Bajaj, Tulasi R. Meeniga, Vidhut Singh, Vijaay Doraiswamy, Vijay Santore, Yogesh Sharma, and Zhelyazko Tumbey for enriching my skill set, technical expertise, and knowledge on software development practices and principles, and for keeping me motivated each and every day.

I am very thankful to the editorial team at Apress and the technical reviewer for having various checkpoints in place and for providing useful feedback in a timely manner, all of which have made this book more useful for you, the reader.

# Introduction

This book is intended to get beginners and intermediate-level software engineers, up and running with API testing, standard coding practices, and the standards and guidelines for better API test automation development and management.

Each chapter starts by explaining the topic it covers, allowing you to skip ahead if you are already aware of the contents.

Chapter 1 introduces APIs, what API testing is, why we need to have API testing during the software development/testing process, types of API testing, and the advantages of testing APIs.

Chapter 2 explains the different architectures used for developing a scalable software web application plus the protocols used for communicating between the client and the server and their attributes.

Chapter 3 talks about different types of authentication used in web-based software applications.

Chapter 4 covers the tools used in API testing: cURL, Postman, and RestAssured. This chapter also has information on the useful frameworks and libraries used in test automation development.

Chapter 5 introduces the test pyramid and why we need to visualize tests on each layer of a software application.

Chapter 6 walks you through the aspects of API testing and the API testing paradigm.

Chapter 7 talks about the components and guidelines for a good test script.

Chapter 8 covers things that are widely missed and never perceived later in the project life cycle phase, but if used will make test automation much better and joyful.

## INTRODUCTION

Chapter 9 talk about the components of the test automation framework and its design aspects. This chapter guides you through writing a test automation framework from scratch.

Chapter 10 is an extension of Chapter 9. In it, you learn how to develop the test script, execute it, and verify the results.

Chapter 11 introduces API documentation developed using the Swagger UI and how to read documentation that will be useful in writing test scripts.

Chapter 12 covers a case study of a shopping cart application of a hypothetical company. A hypothetical character will walk you through the real-life testing working on a Scrum project.

You should have a prior knowledge of the Java programming language and understand the basics of Maven, Tomcat, and Docker. In addition, an awareness of the Spring Framework is good. I use design patterns (Factory pattern, Singleton pattern) and solid design principles in this book so you will gain knowledge on best coding practices.

This book is useful for API testing aspirants and developers/architects. Project managers and non-technical team members will also greatly benefit from reading this book.

The test scripts developed in this book are hosted on GitHub. Any source code or supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at <http://www.apress.com/978-1-4842-8141-3>. For more detailed information, visit <http://www.apress.com/source-code>. For any queries or valuable feedback, feel free to get in touch with me, Jagdeep Jain, at [jagdeep.jain@gmail.com](mailto:jagdeep.jain@gmail.com).

## CHAPTER 1

# Introduction to API Testing

This chapter introduces application programming interfaces (APIs) and API testing. API testing is an important aspect of software testing activities during the development of typical services-based software. It involves testing the application's business components, usually represented as an API, before the UI is developed. A microservice is an API that deals with a single requirement.

By the end of this chapter, you'll have a good idea of the different types of API testing, the need for them, and the advantages of testing at the API level. If you're already familiar with API testing, you may proceed to the next chapter.

## What Is API Testing?

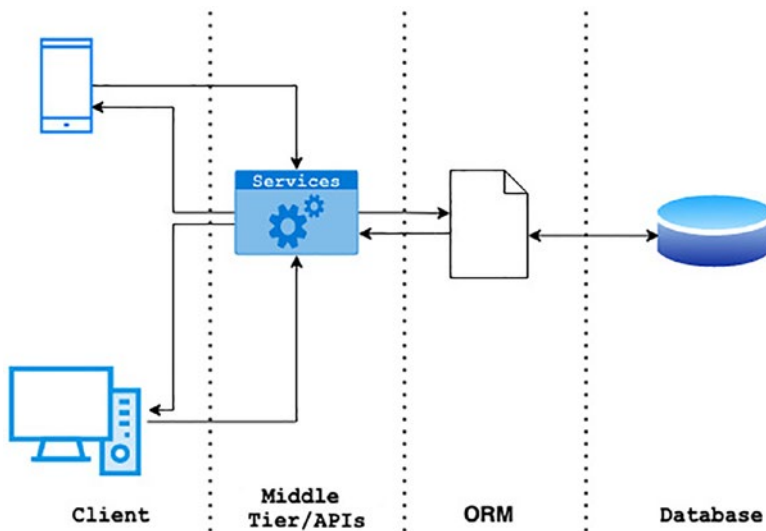
An API abstracts the application layer and provides the resource(s) for consumption by the client. APIs are the backbone of any typical web application, multi-tier web application, or mobile application that hides the inside details of the system, such as how an online payment is processed for a consumer.

APIs are the middle tier of an application and they deal with the back end, usually via an ORM (Object-Relational Mapping) or any other tool, or directly with the database and with the front end. The API acts as an agent

between the back end and the front end. The API reads the data from the back end based on the user requirement/request and sends the response to the front end.

For APIs that do not have a front end, the owner of such an API provides a service-based model to their users, such as a payment gateway, weather forecasting, etc.

Figure 1-1 shows a typical service-based software application architecture. It has a database at the back end, APIs in the middle tier, and requests made from a browser or mobile application. We will discuss this setup in detail in the next chapter.



**Figure 1-1.** *Web-based software application*

A typical web application<sup>1</sup> can be an e-commerce application, where the user wants to see various product offerings and then buy a product as per their needs. Requests are typically made from the front end/GUI. The middle tier has various components in the form of APIs, such as an API

<sup>1</sup>[https://en.wikipedia.org/wiki/Multitier\\_architecture](https://en.wikipedia.org/wiki/Multitier_architecture)