

# Design REST API

Course code: RESTAPI\_DESIGN

Design an API that developers will immediately understand and that will not need to be fixed or versioned immediately after launch. REST APIs are the backbone of virtually every web application today. But there is a huge difference between a working API and a good API. The key to success is to adhere to standards and proven conventions. But how to put them all together so that the whole works synergistically? The key will be to switch to Design First mode and really put API design first. This course will guide you through the world of REST API design from the first EndPoint to a complete set of CRUD and non-CRUD operations. You will learn to design EndPoints that return data in the correct and readable format. We will show you how to enrich the API with standard metadata that developers expect, and how to generate error structures that are understandable and consistent. We will discuss authentication and other mechanisms that you need to know in practice. You don't have to be a programmer. From this training, each participant will take home their own functional REST API, including documentation and a functional mock server.

## What we will teach you

- You will understand the history and all the connections between HTTP and REST
- You will learn to read and implement HTTP standards for REST API
- You will understand key conventions for REST API design
- You will be able to review existing REST interfaces
- You will understand the differences between XML RPC and REST, including the principles of "migration"
- You will learn to design a complete REST API using the ApiDog tool

## Required entry knowledge

- The course is suitable for complete beginners who want to get acquainted with REST API design
- Basic knowledge of how communication works in the web environment is an advantage

## Course outline

### Day 1 - REST API Design

- Introduction to REST, HTTP API and key historical contexts
- Limitations and models: Fielding, Richardson, the idea of HATEOAS
- HTTP as a carrier protocol, working with standards (RFC)
- URL design, filtering, paging and sorting solutions
- HTTP methods, operations, idempotence, domain operations on CRUD
- Return codes and structures, error handling, formatting and standards
- Versioning, HTTP headers, HTTP Cache, Async operations and webhooks

### Day 2 - Design First

- REST API life cycle
- Design First methodology and Open API Specification standard
- Creating OAS specification using various tools
- API mocking, development against specification, validation proxy
- Principle of generating code, libraries and documentation from OAS
- Debugging and testing REST API, tests and monitoring
- ApiDog for designing a complete API including documentation

#### GOPAS Praha

Na Strži 2097/63  
140 00 Praha 4 - Krč  
Tel.: +420 226 201 390  
[info@gopas.cz](mailto:info@gopas.cz)

#### GOPAS Brno

Nové sady 996/25  
602 00 Brno  
Tel.: +420 530 513 590  
[info@gopas.cz](mailto:info@gopas.cz)

#### GOPAS Bratislava

Dr. Vladimíra Clementisa 10  
Bratislava, 821 02  
Tel.: +421 902 903 132  
[info@gopas.sk](mailto:info@gopas.sk)



Copyright © 2026 GOPAS, a.s.,  
All rights reserved