Part 1

Fundamentals of API design

Every journey starts with a first step, and the API design journey is no exception. API designers need many skills and need to take many topics into account when creating APIs, but without a solid foundation, all advanced skills and topics are worth nothing. That is what you'll learn in this first part.

We will first set the scene by explaining what an API is, why it actually must be designed, and what *learning to design an API* actually means. You will discover that although these actually are programming interfaces, APIs are more than "technical plumbing" and that you must learn fundamental principles to design any type of API.

Even before thinking about the programming side, you will see that an API has to be thought of from its users' perspectives. An API is supposed to let your users easily achieve their goals, not the ones of the system exposing the API. Only once these goals are known and accurately described can the actual programming interface, such as a REST API, be designed. And like any programming, describing a programming interface should be done with an adapted tool like the OpenAPI Specification for REST APIs.

1 What is API design?

This chapter covers

- What an API is
- Why API design matters
- What designing an API means

Web application programming interfaces (APIs) are an essential pillar of our connected world. Software uses these interfaces to communicate—from applications on smartphones to deeply hidden backend servers, APIs are absolutely everywhere. Whether these are seen as simple technical interfaces or products in their own right, whole systems, whatever their size and purpose, rely on them. So do entire companies and organizations from tech startups and internet giants to non-tech small and medium-sized enterprises, big corporations, and government entities.

If APIs are an essential pillar of our connected world, API design is its foundation. When building and evolving an API-based system, whether it is visible to anyone or deeply hidden, whether it creates a single or many APIs, design must always be a major concern. The success or failure of such a system depends directly on the quality of the design of all its APIs.