OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○○○○

Applying PBT to OpenAPI /-Generator
○○○

Questions?
○○

Property Based API Testing

EDD Workshop Session

Patrick Pfeifer   Johannes Brügmann

March 6, 2025

OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○○○○

Applying PBT to OpenAPI /-Generator
○○○

Questions?
○○

OpenAPI Basics
●○○○○

Property-Based-Testing Basics
○○○○

Applying PBT to OpenAPI /-Generator
○○○

Questions?
○○

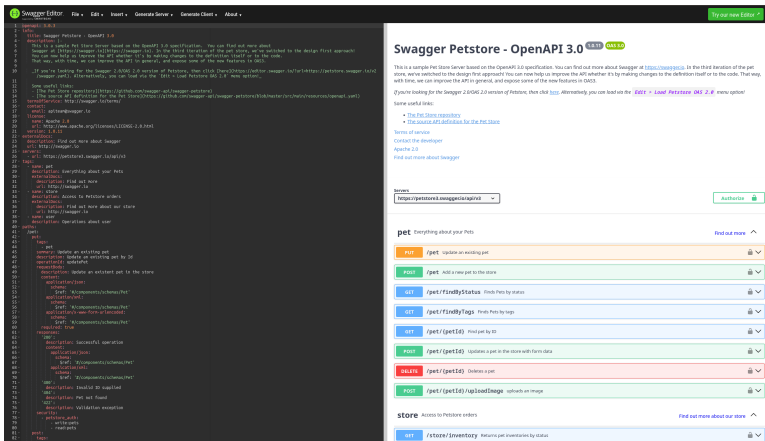# Section 1

OpenAPI Basics

OpenApi: About

- API description language on top of JSON schema
- intended for REST and JSON RPC APIs with `JSON` or `XML` payloads
- language independent API specification written in `JSON` or `YML`
- plenty of generators and tools for individual language bindings
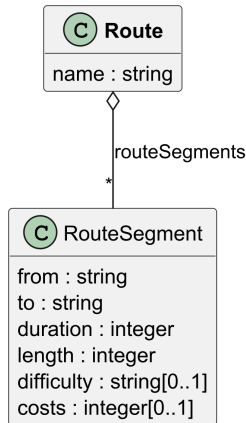- client and server stubs generation for models and api

OpenApi: Building Blocks

- General info block (title, license, server, tags, …)
- Operations (HTTP-verb + Path, request and responses, media-types)
- Schemata (named and anonymous models, parameters)
- Security (OAuth 2.0, HTTP BasicAuth, …)

OpenApi: Swagger Editor

Figure 1: Editieren von OpenApi-Dateien in Swagger Editor.

A1: Modelliere in OpenAPI

Figure 2: Klassendiagramm für das Modell "Route".

OpenAPI Basics
ooooo

Property-Based-Testing Basics
●ooo

Applying PBT to OpenAPI /-Generator
ooo

Questions?
oo

Section 2

Property-Based-Testing Basics

OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○●○○

Applying PBT to OpenAPI /-Generator
○○○

Questions?
○○

PBT: General

Figure 3: Testing Dilemma: "testing can only find bugs, not prove their absence" (E.W. Dijkstra in [1]).

OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○○●○

Applying PBT to OpenAPI /-Generator
○○○

Questions?
○○

PBT: Basics

$$\forall s \in S : P(s) = \top, \qquad P : S \to \{\top, \bot\} \tag{1}$$

- S: state space of valid instances
- P: property to prove, candidates involve [2]:
    - invariants,
    - round-trips,
    - symmetry,
    - commutativity,
    - idem-potency, and
    - induction

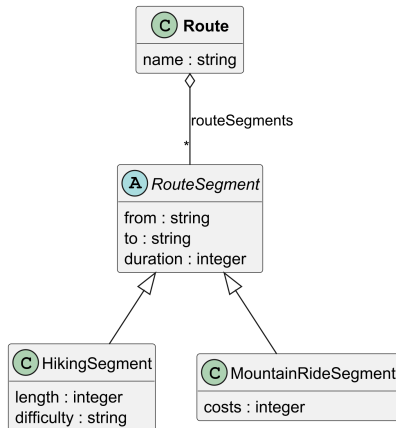# A2: Implementiere PBT `isValid()`

- let `jqwik` generate arbitraries
- validate any arbitraries with `openapi4j`
- use model `Route` from previous exercise

OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○○○○

Applying PBT to OpenAPI /-Generator
●○○

Questions?
○○

Section 3

Applying PBT to OpenAPI /-Generator

OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○○○○

Applying PBT to OpenAPI /-Generator
○●○

Questions?
○○

Folie 3

OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○○○○

Applying PBT to OpenAPI /-Generator
○○●

Questions?
○○

A3: Modelliere in OpenApi und prüfe mit PBT

Figure 4: Klassendiagramm für das polymorphe Modell "RouteSegment".

OpenAPI Basics
○○○○○

Property-Based-Testing Basics
○○○○

Applying PBT to OpenAPI /-Generator
○○○

Questions?
●○

Section 4

Questions?

OpenAPI Basics
○○○○○
Property-Based-Testing Basics
○○○○
Applying PBT to OpenAPI /-Generator
○○○
Questions?
○●

References

[1]   P. Godefroid and K. Sen, "Combining model checking and testing," *Hand-book of Model Checking*, pp. 613–649, 2018, Available: https://patricegodefroid.github.io/public_psfiles/mchandbook2018.pdf

[2]   S. Sahayam, "Property-based testing patterns." Jun. 2016. Available: https://blog.ssanj.net/posts/2016-06-26-property-based-testing-patterns.html