

# APIs mit swagger.io



# Projekt

- eCommerce-Projekt für ERGO Direkt Versicherungen
- Startup Style
- Händler können Versicherungen für ihre Kunden direkt als Upselling im Checkout anbieten
- Einfache Anbindung von Händlern per Plugin, d.h. keine individuelle Programmierung

# Architektur

- ERGO Direkt interne Systeme
- neues Server-Backend mit Anbindung an ERGO Direkt
- Connector-SDK als Client-Backend
- Shop-Plugins die auf das SDK aufsetzen
- Möglichkeit zu Integration in Eigenentwicklungen dank SDK

# Situation

- 1 altes System auf ERGO Direkt Seite
- 3 neue Systeme: Backend, Core, Shop-Plugin
- Kein bestehender API-Code
- Abstimmung zwischen 3 Parteien: 2 Agenturen, Kunde mit eigener Technik
- Fachspezifikation parallel zur Implementierung
- Parallele Entwicklung von Backend und Frontend

# Herangehensweise

- Definition der API mit swagger.io aka Open API Spezifikation
- API-Spezifikation soll selbst versioniert werden
- iterative Updates während Projektlaufzeit
- DRY...für offenen Dialog mit Kunden & Agenturen

# swagger was?

- Open Source (Apache 2.0 License)
- RESTful API
- YAML-Editor mit eingebauter Visualisierung
- Code als Dokumentation
- Swagger-UI für API-Consumer Doku
- Code Generatoren für Client & Server, Mocks, mehr als 20 Sprachen, Community

# Workflow

- Design first:
  - Endpunkte
  - HTTP-Methoden
  - Payloads
- Dokumentation direkt generiert
- Mocks generieren / Beispiel Responses
- Testing direkt aus Editor

# Let's swagger ...



# Lessons learned 1/2

- Editor-Dokument ist ein gutes Kommunikationsmittel mit klarer Struktur
- Wird auch vom Fachbereich verstanden
- Endpunkte nach tags gut filterbar
- Editor-YAML hat Lernkurve bis es validiert ist, zwingt aber zur saubereren Dokumentation & deckt Unschärfen in der Spezifikation auf

## Lessons learned 2/2

- versionierbare API-Spezifikation mit Diffs gerade in der Spezifikationsphase hilfreich
- generiert Doku für Kunden gleich mit (UI)
- CodeGenerator für PHP taugt, kann bei Yaml-Updates neu exportiert werden
- Integrierte Testcalls sehr hilfreich, wenn nicht Full-Stack inhouse entwickelt wird
- Community wächst, z.B. YAML Generierung aus DocBlocks, Swagger-Import in SAAS

# Danke

Stefan Peter Roos

*@spr2*

Tobias Merkl

*@tabsl*