



## Sebastian Blei — freiberuflicher Entwickler

### Anschrift:

Spiegelstraße 27

99734 Nordhausen

☎ +49 1522 334 987 2

✉ mail@sebastianblei.de

---

# Projekte

## 10/2015 — 10/2018

Banking-Partner Beratungsgesellschaft mbH,  
Bad Sachsa

REST-Fullstack:

- Frontend (**ember.js** & ember-cli)
- Backend (**rust** & nickel.rs)
- Continuous Integration (**docker**, travis-ci, packet.net)

Umsetzung: Aufgebaut wurde eine REST-API mit postgresSQL, die das Segmentieren von Kundendaten verschiedener Finanzinstitute berechnete. Dafür wurden die Kundendaten der Finanzinformatik per SQL-Selects abgezogen und bei der API via CSV importiert. Die API beinhaltete Session- & User-Management (jwt), sowie viel Businesslogik zur Segmentierung der Daten. Das Frontend wurde mit ember.js und Bootstrap umgesetzt und diente u.a. zur Administration der

Segmentierungseinstellungen. Session-Management (jwt), Import-Schnittstellen der Daten, sowie grafische Visualisierung der Segmentierungsergebnisse (chart.js), waren ebenfalls wichtige Punkte im Frontend.

## **08/2014 — 10/2015**

CrowdArchitects GmbH, Erfurt

CRUD-Backend:

- API (**node.js & express.js**)
- Datenbank (**neo4j**)

Die CRUD-API, in node.js geschrieben, bot Schnittstellen zur Administration (Frontend in Angular.js) von Konfigurationseinstellungen, sowie Schnittstellen für mobile Apps (Android & iPhone). Die mobilen Apps stellten individualisierte Whitelabel-Apps verschiedenster Radio-Sender dar. Wichtige Schnittstellen, waren u.a. die Ausgabe komplex-verknüpfter neo4j-Nodes an die mobilen Frontends, die sich teilweise über eine tiefe von einem Dutzend Relationen verband. Dafür wurde eine eigens entwickelte ORM-Abstraktion eingesetzt. Die Kommunikation der Frontends mit der API erfolgte via CRUD & Paging, um die benötigten Datenmengen der mobilen Frontends möglichst gering & schnell zu halten.

## **08/2013 — 08/2014**

plazz AG, Erfurt:

- Frontend (**Angular.js**)

Das Frontend diente teilweise zur Konfiguration,

hauptsächlich aber zur Visualisierung informativer Daten aus dem KFZ-Bereich. Die Kommunikation mit dem PHP-Backend erfolgte via REST & Paging.

---

# Erfahrungen

## 8 Jahre

CSS/CSS3 (Sass, Less, etc.)  
HTML/HTML5  
JavaScript (plain / vanilla)  
Templates (mustache.js,  
HandleBars/HTMLBars, E-Mail, Markdown,  
Liquid, jekyll)

## 7 Jahre

vim

## 6 Jahre

bash / zsh / shell  
git

## 5 Jahre

ember.js

## 4 Jahre

node.js  
postgresql / mysql  
REST APIs

## 3 Jahre

rust

## 1 Jahr

CRUD APIs  
TypeScript

## < 1 Jahr

Angular.js

C#

MongoDB

neo4j

Python

react / react.js

Ruby

---