

Inhaltsverzeichnis

"Nur mal kurz" etwas erledigen: (Full) Stack Development	3
Architekturentscheidungen	3
Fachliche und technische Anforderungen	3



17.06.2026

2/4

"Nur mal kurz" etwas erledigen: (Full) Stack Development

"Nur mal kurz" etwas erledigen: (Full) Stack Development

Eigentlich wollte ich nur auf einen Blick sehen, ob meine verschiedenen Geräte und Services gerade online sind. Auch wenn ich unterwegs bin.

So eine Art Monitoring im Heartbeating-Modus wäre doch was dachte ich mir.

Eine entsprechende (statische) Startseite <https://home.jenensch.de> gab es ja auch schon. Allerdings war die bisher nur mittels Bootstrap ein wenig auf responsive gestaltet. Da musste also noch viel passieren. Dieses Wochenende wollten die 10jährigen bei uns Fußball schauen. Der damit verbundene Trubel und die Kinderbespaßung durch Tamara gab mir genug Zeit, Raum und Grund (oder Vorwand?) mit Headset bewaffnet „im Tunnel“ zu verschwinden. Also los...

Architekturentscheidungen

Um sämtliche Services im Blick haben zu können, sollte der Schmalspur-Monitoring-Service sowohl intern als auch extern erreichbar sein, aber sämtliche Services im Überblick haben. Somit entschied ich mich für die Implementierung auf dem Reverse Proxy.

Die Maschine langweilt sich ohnehin die meiste Zeit, da sie selbst nur auf die Server im Netzwerk weiterleitet.

Fachliche und technische Anforderungen

- **Client-Autorisierung:**
Der Backend-Service soll nur von autorisierten Clients (in dem Fall vom eigenen Webclient aus) aufrufbar sein. Hierbei soll der Authentifizierungs-Mechanismus zwischen Frontend und Backend dynamisch und möglichst sicher sein, damit nicht jeder Dödel den Status-Service penetrieren/nutzen kann - durch zeitliche begrenzte „one time tokens“.
- **Caching:**
Der Online-Status der einzelnen Services im Netzwerk soll nur maximal jede Minute abgerufen werden um nicht unnötigen Traffic im LAN zu erzeugen.
- **Portalfunktion:**
Sofern die jeweiligen Services über ein Webinterface verfügen, sollen diese direkt über die Status-Seite aufgerufen werden können. Nicht verfügbare Services sollen als inaktiv bzw. nicht anklickbar visualisiert werden.
- **Responsive Design:**
das Status-Portal soll via Mobile und Desktop betrachtet nicht zu Augenkrebs führen - zumindest nicht sofort.
In dem Zuge gleich die bislang verwendete Bootstrap-Version aktualisieren → 5.2x nach 5.3.8

TODO: Architekturschaubild der (kleinen) Webanwendung, Übersicht der eingesetzten Technologien und Bibliotheken, (Downloadbarer Code)

[api](#), [bootstrap](#), [cors](#), [debian](#), [it](#), [javascript](#), [linux](#), [monitoring](#), [php](#), [public](#), [security](#), [software development](#), [token](#), [responsive](#), [design](#)

Wenn dieser Beitrag für Dich von Mehrwert war, freue ich mich über eine kleine Spende:

From:

<https://wiki.jenensch.de/> - **SJLT**

Permanent link:

https://wiki.jenensch.de/public/articles/fullstack_example

Last update: **01.02.2026**

