

Blockwoche: Web Programming Lab 🚀

Progressive Web Apps

Überblick

Fragen

Diskutiert zu zweit...

> Was sind die Vor- und Nachteile von PWAs ggü. native Apps? (10')

> Optimierung von Kosten, 1 Code Basis mehrere Kanäle
(Web App, Mobile App, Desktop App)

> Für bestimmte Anwendungsfälle nicht geeignet (z.B. Gaming)

> Bessere Visibilität in Suchmaschinen

> Nur beschränkter Hardware Zugriff (einfach das was der Browser zulässt)

> Bessere Performance durch offlinefähiger (lokaler) Content

> Kosteneffektive Aquirierung von User (kein Redirect
in den Store, kein Abgeben von 30%)

Offlinefähigkeit von Web Apps

Motivation und High-Level Architekturüberblick

Offlinefähigkeit von Web Apps

Funktionsweise ServiceWorker und Cache API - Content offline nutzbar machen

Übungsaufgabe (30')

1. Ermittle den Zustand online / offline und gebe diesen aus.
2. Mache den Offline-Seed offline fähig, d.h. der Reload ist auch möglich, wenn der Browser offline ist.

Übung: exercises / online-offline

Starten via «`npm run start`» → `http://localhost:8080`

<https://gist.github.com/bitsmuggler/51e6ce91351162cf250e06b2b888b6>

Offlinefähigkeit von Web Apps

Funktionalitäten einer Web App offlinefähig gestalten mittels IndexedDB

Übungsaufgaben (30')

1. Erstelle ein Formular, das Vor- und Nachname entgegen nimmt und in der IndexDB speichert.
2. Lies die persistierten Personen aus und rendere diese entsprechend in einer Liste.

- Roos Patrick
- Schürmann Andreas

Übung: exercises / indexed-db

<https://gist.github.com/bitmuggler/74fea550ab7847db93e046daa7b3cab2>