

Hochschule für Telekommunikation Leipzig (HfTL)

PROFILIERUNG NETZBASIERTE ANWENDUNGEN

PROFILIERUNG MOBILE APPLIKATIONEN

SOFTWAREDOKUMENTATION

TaskY - Cache und Notifications in mobilen Webanwendungen

David Howon (147102)

Michael Müller (147105)

Wintersemester 2016/17



Hochschule für Telekommunikation Leipzig
University of Applied Sciences

Inhaltsverzeichnis

1	Einleitung	1
2	Anforderungen	2
2.1	funktionale Anforderungen	2
2.2	nicht-funktionale Anforderungen	2
3	Vorbetrachtung	3
3.1	Betrachtung aktueller Technologien	3
4	Realisierung	4
4.1	Architekturbeschreibung	4
4.2	native UI	4
4.3	Umsetzung ServiceWorker	4
4.4	Umsetzung mittels nativem Android Service	4
5	Abgrenzung	5

1 Einleitung

Diese Dokumentation entstand im Rahmen der Profilierungen „Mobile Applikationen“ und „Netzba-sierte Anwendungen“ im Wintersemester 2016/17 an der Hochschule für Telekommunikation Leipzig (HfTL).

... Einleitung moderne webtechnologien -> webapps statt nativen Apps ...

... Beschreibung der Aufgabe/des Problems ...

... Versuch der Lösungsfindung/Kurzbeschreibung Projekt ...

2 Anforderungen

2.1 funktionale Anforderungen

Die WebApp soll für den **Mehrbenutzerbetrieb** ausgelegt werden. Ein Benutzer soll sich für die Nutzung **Registrieren** und anschließend am Portal **Anmelden** können. Es können eigene **Aufgabenlisten** angelegt, bearbeitet oder gelöscht werden. Weiterhin können Aufgabenlisten mit mehreren Benutzern (Kontakte bzw. Gruppen) geteilt werden.

Zu einer Aufgabenliste können einzelne Aufgaben zugeordnet werden. Eine Aufgabe muss mindestens aus einem Titel bestehen und kann mit einem Ort, einer Beschreibung, einer hinterlegten Checkliste, einem Zeitraum sowie einer Fälligkeit erweitert werden.

Der Benutzer soll mittels PUSH-Benachrichtigungen über aktuelle Änderungen an den Aufgabenlisten und Gruppen informiert werden.

2.2 nicht-funktionale Anforderungen

- Look and Feel einer nativen Android App
- Single Page Application (SPA)

3 Vorbetrachtung

3.1 Betrachtung aktueller Technologien

3.1.1 UI-Technologien

Betrachtung und Bewertung unterschiedlicher Möglichkeiten, die Anforderungen an das Benutzerinterface zu erfüllen.

3.1.2 Serverlogik und Schnittstellen

Welche Technologien sollen auf Serverseite eingesetzt werden und warum?

4 Realisierung

... Wie wurden die beiden Ideen umgesetzt ...

4.1 Architekturbeschreibung

... Beschreibung (mit Schema) der Softwarearchitektur ...

4.2 native UI

Um das „Look and Feel“ einer nativen App zu erreichen wird das UI-Framework **nativeDroid2** verwendet.

... Beschreibung, welche Views vorhanden sind ...

4.3 Umsetzung ServiceWorker

... Umsetzung mittels ServiceWorker ...

4.4 Umsetzung mittels nativem Android Service

... Umsetzung mittels eigenem nativen Service ...

5 Abgrenzung

... Was kann nicht geleistet werden? ...

... Was ist eventuell zukünftig möglich ? ...