

Mobile Anwendungen

Dokumentation

Gruppe M

Raphael Barth Mike Sickmüller Marius Cerwenetz

Mannheim, Sommersemester 2020

Inhaltsverzeichnis

1	Beschreibung	1
2	Architektur	1
3	PDF	1
4	RoomLifecycleService	1
5	Tag - QR und NFC	2
6	MQTT Service	2
7	Datenbank	2
8	Fazit	2
9	Ausblick	2

1 Beschreibung

2 Architektur

3 PDF

4 RoomLifecycleService

Der RoomLifecycleService steuert, wie der Name bereits andeutet, in welchem Zustand sich Räume befinden. Er wird ausschließlich im Host-Modus betrieben. Die Entscheidung fiel auf einen Service, weil wir auch im nicht-geöffneten Zustand der App auf Nachrichten von außen reagieren wollten. Beispiel: Der Host eröffnet einen Raum und wechselt die App oder sperrt das Smartphone. Wenn die Kontaktverfolgungssapp nun nicht mehr im Vordergrund ist während das Timeout ausläuft oder der Raum in die Öffnungszeit reinrutscht, dann würde er nicht geschlossen respektive geöffnet werden. Um also zu vermeiden dass der Host über die gesamte Zeit sein Smartphone nicht verwenden kann und das Display mit hohem Stromverbrauch den Akku entleert haben wir uns für einen Service entschieden.

Wird der Service gestartet wird die Raumüberprüfungsroutine nebenläufig in einem neuen Thread gestartet. Dadurch verhindern wir dass das der Hauptthread machen muss. Der Raumstatus wird jede Sekunde mit einem Busy-Wait-Algorithmus überprüft. Das sollte man nicht in den Hauptthread auslagern, da er diesen unnötig blockiert. Der Datenbankzugriff ist nicht die Ursache für die Designentscheidung, denn dadurch dass wir schon über das Repository nebenläufig auf die Datenbank zugreifen sind wir bezüglich Datenbank-Latenz bereits auf der sicheren Seite. Im Thread überprüfen wir initial ob aktuell geschlossene Räume geöffnet und aktuell geöffnete Räume geschlossen werden sollten. Nebenher wird im Mainthread versucht den MQTT-Service zu einzubinden. Denn der Host soll natürlich auf alle Räume die sich öffnen hören bzw. aufhören auf alle Räume zu hören die sich geschlossen haben. Außerdem muss der Host den Teilnehmern die Raumeigenschaften und den Öffnungsstatus beim Öffnen mitteilen. Wir brauchen den MQTT-Service hier also um Raumstatusänderungen den Teilnehmern mitzuteilen. Zurück zum LifeCycleService. Ein Raum hat drei mögliche Zustände:

1. Wird sich öffnen
2. Geöffnet
3. Geschlossen

Jede Sekunde wird also überprüft ob Räume die geschlossen sind geöffnet werden sollen, ob Räume die sich öffnen werden geöffnet werden können und ob Räume die geöffnet sind geschlossen werden sollten. Falls noch Teilnehmer in zu schließenden Räumen sind werden Sie rausgeworfen. In der Datenbank wird dann für alle Teilnehmer als Austrittszeitpunkt die Timeout-Zeit des Raums eingetragen. Auf der Teilnehmerseite reagiert

der Observer auf die Änderung des Raumstatus und beendet den Raum auch real für den Teilnehmer.

5 Tag - QR und NFC

6 MQTT Service

7 Datenbank

8 Fazit

9 Ausblick