

caption

# **Inhaltsverzeichnis**

## **1 Evaluation**

**1**

# 1 Evaluation

In diesem Kapitel ...

## Verwalten des Zustands

Durch den 'StateManager' Klasse ist diese Anforderung zu großen Teilen abgedeckt. Ein neuer Zustand kann ausschließlich indirekt unter sicheren Bedingungen gesetzt werden. Des weiteren steht Funktionalität für den Zugriff auf den Zustand zur Verfügung, ohne dabei eine Änderung am Original vornehmen zu können. Ein interessanter Zusatz wäre die Option den Zustand observieren lassen zu können und jedem Beobachter über eine Änderung zu informieren. Damit bestände die Möglichkeit, durch eine Modifikation am Zustand ohne größeren Aufwand anderweitige Logik ausführen zu lassen. Dadurch würde das Framework ebenfalls zusätzlich an Reaktivität gewinnen.

**Überprüfung des Zustands als unveränderliche Datenstruktur** Mittels der angewendeten Reflexion ist dieser Punkt im Großen und Ganzen erfüllt. Ein bessere Lösung stellt jedoch die Generierung einer unveränderlichen Zustands Klasse (hier 'data' Klasse ) mittels Annotation dar. Damit ginge man einer Überprüfung zur Laufzeit und dem damit einhergehenden Problemen (keine Typsicherheit etc.) aus dem Weg und stellt von Anfang an die Korrektheit des Zustands sicher.

## Abbildungsverzeichnis

## Online References

- [1] David Karnok. *Writing operators for 2.0*. 8. Mai 2018. URL: <https://github.com/ReactiveX/RxJava/wiki/Writing-operators-for-2.0> (besucht am 21.08.2019).