



---

## SOFTWARE-ARCHITEKTUR:

### AUFGABENPAKET 3

---

20. Februar 2022

### Vorbemerkung

Die Hauptthemen dieses Arbeitspakets sind nochmal „Verteilte Systeme“. Außerdem behandelt es eine praktische Aufgabe zum MVC-Pattern. Vor Bearbeitung der restlichen Übungen im Skript sollen Sie die Abschnitte zum Thema Microservices 4.1.2.5 und 4.2.8 lesen. Besonders interessant sind die Abschnitte 4.2.8.2 (API-Gateway), 4.2.8.4 (Health Check) und 4.2.8.5 (Management-Services). Details zum MVC-Pattern finden Sie im Skript in Abschnitt „Interaktive Systeme“ unter Punkt 4.1.3.1.

### Theoretischer Teil

#### 1. Paper zum Thema Microservices

Microservices sind zur Zeit in aller Munde. Neben einem Abschnitt im Skript finden Sie im Übungspaket zwei Artikel zu diesem Thema:

1. Alexander Schwartz: MICROSERVICES – MEHR ALS NUR EIN HYPE? [[Sch17](#)]
2. Henning Sternkicker und René Meyer: MICROSERVICES – HYPE ODER SEGEN? [[SM16](#)]

Verfassen Sie ausgehend vom Skript und den beiden Artikeln ein *One Page Paper* über das Thema „Microservices“, in dem Sie in das Thema einführen. Sie dürfen natürlich weitere Literatur hinzuziehen. Das Paper richtet sich an den durchschnittlichen IT-Analysten-Studenten zu Beginn des Studiums. Sie können somit davon ausgehen, dass dem Leser gewisse IT-Begriffe bekannt sind, das Thema „Microservices“ allerdings nur als Schlagwort. Der Inhalt muss auf eine einfache Seite passen, wobei die Schrift nicht kleiner als 11pt gewählt werden darf. Überlegen Sie, was wichtig ist und was eher nicht. Sie werden sehen, dass das gar nicht so einfach ist.

- (a) (10 Punkte) Paper wurde abgegeben und entspricht den Anforderungen.

## Praktischer Teil

Nutzen Sie eclipse oder IntelliJ zum Bearbeiten der nachfolgenden Aufgaben. Laden Sie Ihre Lösung als ZIP-Datei hoch. Build-Artefakte sollten nicht hochgeladen werden.

### 2. Aufgabe: MVC-Muster

Abbildung 1 zeigt den schematischen Aufbau des MVC-Patterns. Die wesentliche Idee bei dem Pattern ist, dass das Modell von der Eingabe bzw. Anzeige abgekoppelt wird. Das Modell enthält keine expliziten Referenzen auf den Controller oder die View.

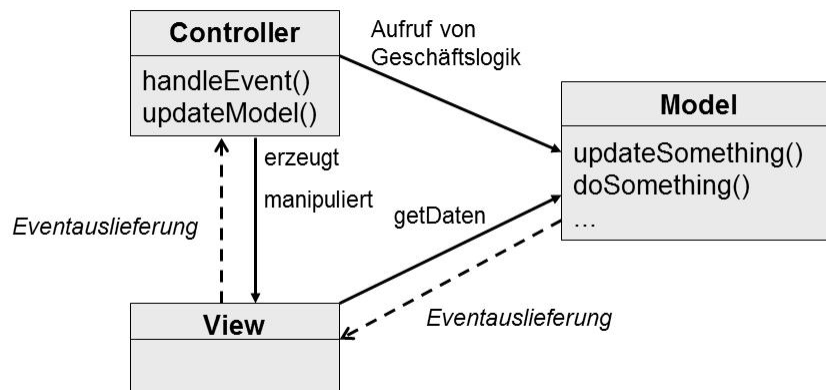


Abbildung 1: Der Aufbau des MVC-Patterns.

Im Übungspaket finden Sie eine kleine Anwendung für einen *Ticker* (siehe Abbildung 2). Die Anwendung besteht aus einem Ticker, der nach dem Start alle 100 Millisekunden seinen internen Zähler um Eins erhöht. Die Buttons sind in der Ausgangsversion noch ohne Funktionalität. Die Anwendung kann über eine Eingabe auf der Konsole gesteuert werden.



Abbildung 2: Die View des Counters.

Zur Steuerung der Anwendung stehen folgende Befehle zur Verfügung: `start` – startet den internen Zähler des Tickers, `stop` – stoppt den internen Zähler, `reset` – setzt den internen Zähler auf Null zurück. Über `show` wird der aktuelle interne Zähler des Tickers auf dem Display angezeigt und mit der Eingabe von `help` werden alle möglichen Befehle aufgelistet. Durch die Eingabe von `exit` oder durch das Schließen des Fensters wird die Anwendung beendet (siehe Abbildung 3).

Im Moment übernimmt die Klasse `TickerTest` die Steuerung der Anwendung. In einer Endlosschleife wird immer wieder die Konsoleneingabe abgefragt. Modifizieren Sie das

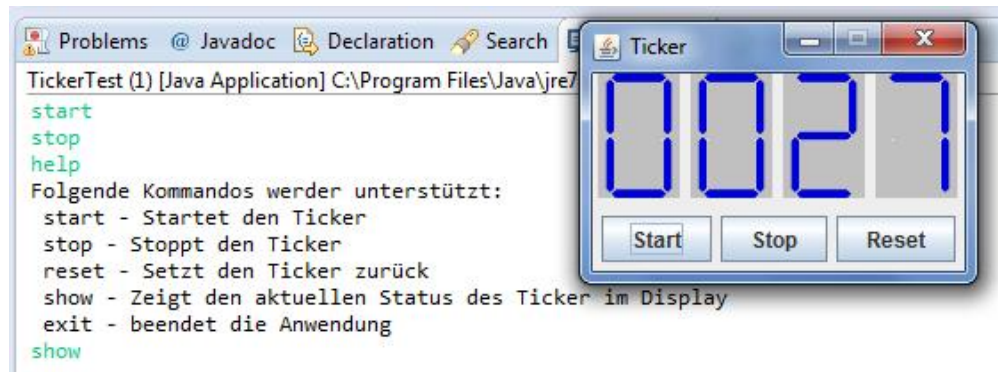


Abbildung 3: Steuerung des Tickers über die Konsole (aus Eclipse heraus).

Programm so, dass es dem MVC-Pattern entspricht. Die Anwendung soll über die Buttons gesteuert werden und auf dem Display soll immer fortlaufend der aktuelle Wert des Tickers angezeigt werden, ohne dass eine explizite Befehlseingabe dafür erfolgt.

- (a) (5 Punkte) Die Anwendung funktioniert, ist korrekt modifiziert und nutzt das MVC-Pattern wie in Abbildung 1 dargestellt.

## Literatur

- [Sch17] SCHWARTZ, Alexander: Microservices – Mehr als nur ein Hype? In: *Informatik Spektrum* 40 (2017), Nr. 6, S. 590–594
- [SM16] STERNKICKER, Henning ; MEYER, René: Microservices – Hype oder Segen? In: *OBJEKTspektrum* Ausgabe Microservices und Docker (2016)