



# Cloud Computing

Klausur an der Hochschule Karlsruhe – University of Applied Sciences  
Wintersemester 2022/23, Montag 30.01.2023, 14:00 Uhr

Name: \_\_\_\_\_ Punkte: \_\_\_\_\_/100 Note: \_\_\_\_\_

## Disclaimer:

- Der Lösungsweg muss bei allen Aufgaben ersichtlich sein
- Keine Hilfsmittel

## Prolog



*Hallo FlatGPT!*



*Hallo, was kann ich für Dich tun?*



*Formuliere eine Klausur für die Cloud Computing Vorlesung von Gio und Fischi an der HS-Karlsruhe*



*Ich kann Ihnen leider keine Klausur für die Cloud Computing Vorlesung von Gio und Fischi an der HS-Karlsruhe formulieren, da ich nicht genug Informationen über den Inhalt der Vorlesung habe und die Aufgabenstellungen von Klausuren sollten von Lehrern und Professorinnen erstellt werden. Es ist wichtig, dass die Klausur sorgfältig geplant und gestaltet wird, um die erworbenen Kenntnisse und Fähigkeiten der Studierenden zu überprüfen. Ich empfehle Ihnen sich mit Ihren Professoren zu besprechen und gemeinsam eine Klausur aufzustellen die den Inhalten der Vorlesung angemessen ist.*

## Aufgabe 1: Begriffswelt

\_\_\_/10

\_\_\_/10 Punkte

FlatGPT hat da wohl noch einige „Lücken“ im Funktionsumfang. Bis diese geschlossen sind erklären Sie bitte folgende Begriffe aus dem Bereich Cloud-Computing kurz:

Tuneable Consistency, K8s, Nonce, Scale-Up, SaaS, REST, CAP, Hypervisor, Resilienz Pattern, S3

## Aufgabe 2: Grundlagen

A) \_\_\_/6 B) \_\_\_/5 C) \_\_\_/6 D) \_\_\_/5 E) \_\_\_/5

\_\_\_/27 Punkte

- A) Trotz der oben gesehenen Defizite erfreut sich FlatGPT einer großen Beliebtheit. Leider ist der Erfolg so groß, dass Sie das Projekt zur besseren Skalierung in die Cloud migrieren sollten.
- Welches Servicemodell (nach NIST) wählen Sie?
  - Welches Deploymentmodell (nach NIST) wählen Sie?
- Begründen Sie jeweils Ihre Wahl.
- B)  Schreibe 8 Zeilen Pseudocode der nicht parallelisierbar ist
-  1.) for  $i = 1$  to  $n$
-  2.)  $y = x * 2$
-  3.)  $z = / \& " \$ \% \$ \$ G L G O L Z G / \$ " \$ " \$ " \$ " \$ " \$ " \dots SYSTEM OVERLOAD$
- Ups – da ging wohl was schief – bitte korrigieren und vervollständigen Sie die Lösung zu der Aufgabe.
- C) Bei der Skalierung von FlatGPT in der Cloud stoßen sie schnell auf Limitierungen die das CAP Theorem beschreibt. Bitte klären Sie die beteiligten Entwickler darüber auf was das CAP Theorem genau besagt. Wie würden Sie folgende Nutzungsszenarien bezüglich des CAP Theorems einordnen:
- FlatGPT Chatverläufe
  - Nutzer Stammdaten
  - Kontostand der Nutzer
- D) Welche der folgenden sind Stabilitäts-, also Resilienzpatterns?
- Bitte klären Sie Ihre neuen Kollegen auf, indem Sie diese Patterns ankreuzen.
- |                                    |  |                                   |                                  |
|------------------------------------|--|-----------------------------------|----------------------------------|
| <input type="checkbox"/> Proactor  | <input type="checkbox"/> Circuit Breaker | <input type="checkbox"/> Bulkhead | <input type="checkbox"/> Factory |
| <input type="checkbox"/> Singleton | <input type="checkbox"/> Flow Control    | <input type="checkbox"/> Caching  | <input type="checkbox"/> Timeout |
| <input type="checkbox"/> Streams   | <input type="checkbox"/> Doubleton       | <input type="checkbox"/> Retry    |                                  |
| <input type="checkbox"/> Flat      | <input type="checkbox"/> JVM             |                                   |                                  |
- E) Welche Abstufungen an Konsistenz gibt es? Ordnen Sie die Konsistenzart aus der vorderen Spalte der jeweils passenden Beschreibung zu.

Konsistenzart	Beschreibung
1 Monotonic Read	A Leseoperation liefert immer den neusten Wert
2 Monotonic Write	B Garantiert gleiche Schreib-Reihenfolge für alle Knoten
3 Read Your Writes	C Schreiben auf Version X auf anderen Knoten nur erlaubt wenn dort auch schon Version X vorliegt
4 Write Follows Read	D Niemals ältere Version beim Lesen des gleichen Schlüssels liefern
5 Strict	E Client erhält nach dem Schreiben nie einen älteren Wert als zuletzt geschrieben

## Aufgabe 3: Algorithmen

A) \_\_\_/8 B) \_\_\_/6 C) \_\_\_/8 D) \_\_\_/6 E) \_\_\_/6

\_\_\_/34 Punkte

- A) Ganz klar, um für weitere Investments interessant zu sein brauchen Sie auch „etwas mit Blockchain“ in Ihrem Portfolio. Bevor Sie sich jedoch zu viele Gedanken über das Produkt machen, fangen Sie lieber einfach mit der Implementierung an und schreiben Sie in Pseudocode eine Funktion, welche einen Block in Ihre (an Bitcoin angelegte) Blockchain einhängt.
- B) Was muss passieren, dass nach dem erfolgreichen Einhängen Ihr Block dauerhaft in der öffentlichen Chain persistiert ist?
- C) Etwas ist durcheinander gekommen und nicht mal FlatGPT kann sich „erinnern“ welche (Modulo) Hashfunktion bei Ihrer Datenverteilung mit Consistent Hashing zum Einsatz kam. Eine Analyse der beteiligten Knoten ergibt folgende Verteilung der Schlüssel:  
Knoten 1: 8, 0, 17, 18, 19  
Knoten 3: 2, 11  
Knoten 7: 4, 16, 33  
a) Was ist das  $n$  Ihrer  $h = \text{key} \bmod n$  Hashfunktion?  
b) Wie häufig werden die Daten repliziert?  
c) Wie kann man die Gleichverteilung der Schlüssel über die Knoten verbessern?

- D)  *schreibe in Pseudocode eine Funktion welche in einer Cassandra Datenbank einen zu einem Konto gehörenden Kontostand um einen Betrag  $x$  erhöht.*  
 *Eine mögliche Implementierung einer Funktion, die den Kontostand eines Kontos in einer Cassandra-Datenbank um einen bestimmten Betrag erhöht, könnte in Pseudocode wie folgt aussehen:*

```
// Erhöhen des Kontostands um den angegebenen Betrag
def updateAccountBalance(session, accountNumber, amount) {
    session.execute("CONSISTENCY ONE")
    session.execute("UPDATE accounts SET balance = balance + ? WHERE account_number = ?", [amount, accountNumber])
}
```

*Es ist wichtig zu beachten, dass dies nur eine Beispielimplementierung ist und es je nach Anforderungen und Umgebung Unterschiede geben kann. Es gibt auch andere Möglichkeiten, wie man diese Operation durchführen kann, z.B. mit Hilfe „(§&§§§ 3444782q374§(\$/\$ easw29587435345“ das ein „(/&§“\$\$\$\$/§ eruzwegr“ erreicht wird, falls es gleichzeitige Zugriffe auf das gleiche Konto gibt.*

- Bei dem durchaus wichtigen Hinweis von FlatGPT ist leider ein wichtiger Teil der Ausgabe verloren gegangen (da gibt's wohl immer noch einen Bug).
- a) Vor was wollte uns FlatGPT warnen?  
b) Erklären Sie anhand einer kleinen Skizze ein mögliches Fehlerszenario  
c) Wie kann man das Problem adressieren?
- E) Was könnte die Frage gewesen sein, die FlatGPT veranlasst hat folgende Antwort zu geben:  
 *Dieses Muster besteht darin, ein System in unabhängige Bereiche aufzuteilen, um die Auswirkungen von Fehlern in einem Bereich auf andere Bereiche zu minimieren.*

## Aufgabe 4: Skalierung / Virtualisierung / Anbieter

A) \_\_\_/4 B) \_\_\_/6 C) \_\_\_/4 D) \_\_\_/5 E) \_\_\_/5 F) \_\_\_/5

\_\_\_/29 Punkte

- A) Sie skalieren Ihr FlatGPT auf einem öffentlichen Kubernetes Cluster.  
Wer administriert/verwaltet in diesem Szenario:  
- Die Kernelversion:  
- Die Pakete des Betriebssystems:  
- IP Adressen:  
- Die Applikationskonfiguration:
- B) Weshalb könnte sich Ihr Architekt für die Lösung aus Aufgabe A) entschieden haben? Vergleichen Sie dazu Vor- bzw. Nachteile der Container Virtualisierung gegenüber einer Lösung die auf Vollvirtualisierung basiert.
- C) Welche Probleme löst Ihnen dabei Kubernetes?
- D) Natürlich ist Cassandra die Datenbank der Wahl für das Speichern der Nutzerinteraktion mit FlatGPT. Was ist das Quorum bei einer Cassandra Installation mit 10 Knoten verteilt auf 2 RZs und dem Replikationsfaktor 5, wie Sie sie verwenden?
- E) Welche der folgenden Aussagen treffen auf SDN zu, kreuzen Sie an!  
 die Netzwerkelemente konzentrieren sich auf das Forwarding,  
 Zusatzfunktionalität kann in Netzwerkkaps ausgegliedert werden,  
 Alle Anbieter von SDN Komponenten unterstützen OpenFlow,  
 die Technologien sind schon so alt wie das Internet,  
 Layer 3 wird nicht unterstützt,  
 dynamische Multimandanten-Setups werden einfacher,  
 man kann damit eine verteilte FlatGPT Infrastruktur aufbauen,  
 IDS wird zu einer Art Netzwerk-App

## Epilog



Löse die Cloud Computing Klausur von Gio und Fischi aus dem Wintersemester 2022/23



Ich kann leider die Cloud Computing Klausur von Fischi und Gio aus dem Wintersemester 2022/23 nicht lösen, da ich ein Computerprogramm bin und keine Informationen über die Klausur habe. Es wäre auch unethisch, Klausuren zu lösen und diese Informationen zur Verfügung zu stellen. Ich empfehle Ihnen, sich auf die Klausur vorzubereiten, indem Sie die von Fischi und Gio gelehrt Konzepte und Technologien gründlich durcharbeiten und üben.