

Portfolio-Prüfung zur Vorlesung "Verteilte Systeme"

Kurs: WWI21SEA – 4. Semester

Die Prüfung setzt sich aus einem Programmierprojekt und einer Demo-Vorführung zusammen. Die Aufgabe soll in Dreier- oder Vierergruppen gelöst werden. Es werden Einzelnoten vergeben. Je nach Komplexität und Qualität der implementierten Lösung können unterschiedlich viele Prozentpunkte (max. 100 Prozentpunkte; entspricht 1,0) erreicht werden:

- Teil 1: max. 60 Prozentpunkte
- Teil 2: max. 40 Prozentpunkte

Aufgabenstellung: Teil 1

Es ist mit Java ein TCP-basiertes Chat-System bestehend aus Nutzeranwendung und Server zu implementieren, wobei für die Basisversion nur die folgenden Java-Imports verwendet werden dürfen: `java.io.*`; `java.net.*`; `java.util.*`; ggf. `java.swing.*`

Auf der Nutzer-Seite soll sich ein Chatnutzer bei einem Server anmelden und dann soll er über einen Server zu genau einem anderen Nutzer einen Chat beginnen und betreiben können.

Aus Fehlertoleranzgründen soll es zwei identische Server auf unterschiedlichen Rechnern im Netzwerk geben. Beide Server müssen den identischen kompletten Chat-Verlauf besitzen.

Ein Chatnutzer kontaktiert einen zufällig ausgewählten Serverrechner (\Rightarrow Lastverteilung) als Anlaufstelle, um sich zentral anzumelden (z. B. Kennung und Passwort) und um einen Chat zu betreiben. Es sollen mindestens drei unterschiedliche Nutzer verwaltet werden können.

Eine Textnachricht, die ein Benutzer in den Chat einfügt, soll im eigenen Chatverlauf angezeigt, zeitnah über einen Server an den Chat-Partner gesendet und in dessen Chat-Verlauf angezeigt werden, sobald dieser bei einem der Server angemeldet ist. Alle Chatnachrichten sollen in korrekter Reihenfolge gespeichert und angezeigt werden.

Der Server muss in der Lage sein, die Chats der Nutzer immer zeitnah verarbeiten zu können, daher soll er als *non-persistent Server* konzipiert werden.

Das Chatsystem muss mit Ausfällen und Kommunikationsfehlern adäquat umgehen können, wobei Sie davon ausgehen dürfen, dass Netzwerkpartitionierungen selten und kurz sind.

Ein Testszenario, das auch Ausfälle und Fehlersituationen simuliert, ist zu implementieren und vorzuführen und als Videodatei einzureichen.

Aufgabenstellung: Teil 2

Aus Gründen der Fehlertoleranz soll das System aus Teil 1 nun mit drei replizierten Server und der Majority Consensus Strategie (MCS) als Replikationskontrollstrategie implementiert werden.

Beschreiben Sie, wie das neue System funktioniert und welche Erweiterungen vorgenommen werden müssen. Implementieren Sie das System inklusive eines Testszenarios, das auch Ausfälle und Fehlersituationen simuliert. Demonstrieren Sie Ihr System und das Testszenario in einem Video.

Abgaben

Pro Gruppe ist eine schriftliche Ausarbeitung zu Entwurf, Implementierung und Testdokumentation ist zu erstellen (ca. 10 Seiten plus kommentiertes Code-Listing) und einzureichen. Dabei ist zu dokumentieren, welches Gruppenmitglied welche Leistungen erbracht hat. Ebenso sind der Java-Quellcode sowie lauffähige class-Dateien mit einer kurzen Bedienungsanleitung einzureichen.

Große Videodateien dürfen ggf. in einer Cloud gespeichert werden. Der Link zum Abruf muss dann in der Dokumentation angegeben werden. Achten Sie dabei auf ausreichende Zugriffsrechte.

Eine von allen Gruppenmitgliedern persönlich unterschriebene ehrenwörtliche Erklärung mit folgendem Wortlaut ist ebenfalls mit einzureichen:

*Hiermit erklären wir ehrenwörtlich, dass wir die vorliegende Portfolio-Arbeit zur Vorlesung "Verteilte Systeme" bestehend aus Ausarbeitung, Programmcode und Video selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben.
(Datum u. Unterschriften)*

Im moodle-Kursraum wird eine Abgabemöglichkeit eingerichtet.

Video

Die Demo-Vorführungen sind als Video mit gesprochenen Kommentaren aufzunehmen. Die beiden Teile sollen dabei nacheinander vorgeführt werden.

Projektbeginn: 22. März 2023

Abgabe der Dokumentation: spätestens 23. April 2023, 23:59 Uhr