*Verteilte Systeme im Sommersemester 2023*

Luca Schöneberg, Matr. Nr. 979049

Henry Wulff, Matr. Nr. 1014530 Osnabrück, 28.3.2023

# Aufgabenblatt 2 - Realisierung eines einfachen Web-Servers

*In dieser Aufgabe sollen Sie einen einfachen Web-Server auf Basis von TCP-Sockets implementieren. Dazu muss Ihr Programm eine Teilmenge des HTTP-Protokolls implementieren.*

# Einleitung

In dieser Übung wurde ein einfacher Web-Server auf Basis von TCP-Sockets implementiert, der eine Teilmenge des HTTP-Protokolls verwendet. Der Server unterstützt GET-Requests und behandelt Dateitypen wie HTML und JPG. In einer freiwilligen Zusatzaufgabe wurde der Server erweitert, um POST-Anfragen zu verarbeiten und Formulardaten zu übertragen.

# Implementierung

Der Web-Server wurde wie folgt implementiert:

## Mehrstufiger Server

Der Server wurde als mehrstufiger Server konzipiert, um mehrere gleichzeitige Anfragen verarbeiten zu können.

## Starten des Servers

Der Server wird mit Angabe eines Ports und eines Verzeichnisses gestartet, in dem HTML-Dokumente gespeichert sind: HTTPserv <docroot> <port>

## Verwendung von Port 8080 oder 8008

Da Port 80 spezielle Privilegien erfordert, wurde Port 8080 (oder alternativ 8008) verwendet.

## Verarbeitung von GET-Requests

Der Server unterstützt die Verarbeitung von GET-Requests. Bei Anfragen nach einer Datei wird diese aus dem Arbeitsverzeichnis gesucht, gelesen und an den Client zurückgegeben (Statuscode 200). Bei nicht gefundenen Dateien wird der Statuscode 404 zurückgegeben. Bei Anfragen nach einem Verzeichnis wird dieses in Form eines HTML-Dokuments ausgegeben, in dem die vorhandenen HTML-Dateien und Unterverzeichnisse verlinkt sind.

Hier ist ein Codeausschnitt für die Verarbeitung von GET-Requests:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## Behandlung von Dateitypen

Der Server behandelt Dateitypen wie Bilddateien im JPG-Format korrekt. Der Content-Type im Response-Header wird angepasst und die Dateien werden binär übertragen.

## Freiwillige Aufgabe POST

Der Server soll nicht nur die GET Requests beantworten können sondern auch spezifische POST- Requests. Es wurde implementiert, dass der Server zwei Parameter entgegennehmen kann und diese multipliziert. Es fehlt jedoch das Senden des POST Request mit dem Anfügen der Parameter im Body.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

## Testen mit verschiedenen Browsern

Der Server wurde mit mehreren Browsern getestet:

Safari vs Firefox:

Die Dividier von Safari fallen in der Anzeige dünner aus als die bei Firefox.

Die Farben des Bildes haben einen Leichen Unterschied in der Graustufe.

Bei den Formularen bestehen in Bezug auf die Buttons & Formular Felder ein Unterschied in der Darstellung.

Safari vs Brave:

Bei den Formularen bestehen in Bezug auf die Buttons & Formular Felder ein Unterschied in der Darstellung.