

## Projekt Systementwicklung SS 2017 - Aufgabe 1

### **Vorbereitungen für das Praktikum:**

Lesen Sie den HTTP 1.0 Standard (RFC 1945). Sie müssen zum Beginn des Praktikums die Terminologie (Abschnitt 1.2) von HTTP 1.0 grundlegend verstehen und HTTP Requests (besonders Abschnitte 5.1.1, 5.1.2, 5.2) und HTTP Responses (besonders Abschnitte 4, 6, 9) verstehen.

Lesen und verstehen Sie weiterhin die Praktikumsaufgaben 1 bis 3 und überlegen Sie sich, welche Schritte nötig sind, um die Praktikumsaufgaben zu lösen.

### **Projektplan**

Erstellen Sie bis zum Ende des Praktikums eine erste Liste der zu erledigenden Arbeitspakete („Projektplan“) aus den Aufgaben 2-3. Pro Praktikumsaufgabe ist es durchaus angebracht, 20-30 Arbeitspakete aufzulisten. Verteilen Sie die Arbeitspakete dann auf die Gruppenmitglieder und versehen Sie sie jeweils mit einer Deadline, zu welcher das jeweilige Arbeitspaket fertiggestellt werden muss.

Wählen Sie weiterhin einen Projektleiter, der den Projektfortschritt überwacht. Es liegt in der Verantwortung des Projektleiters während des gesamten Kurses dafür zu sorgen, dass die Arbeitspakete zeitgemäß fertig gestellt werden. Ist es absehbar, dass das verantwortliche Gruppenmitglied das Arbeitspaket nicht zeitgemäß fertig stellt, dann muss der Projektleiter dieses Arbeitspaket entsprechend um planen.

Sprechen Sie frühzeitig mit Prof. Schinzel, wenn absehbar wird, dass Gruppenmitglieder auch nach mehrfacher Aufforderung nicht den gestellten Aufgaben nachkommen.

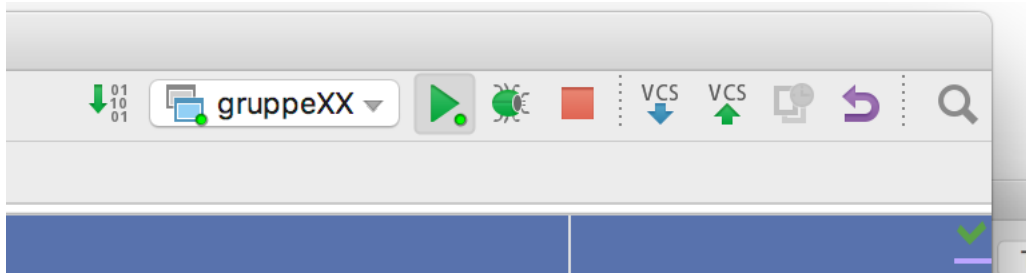
## Arbeitsplatz einrichten

Sie finden im Ilias Videos, die beschreiben, wie Sie Ihr Projekt in Clion erstellen und ins Gitlab übertragen. Führen Sie weiterhin die folgenden Schritte durch:

1. Wenn alles geklappt hat, haben Sie jetzt ein kleines „Hallo Welt!“-Programm im Gitlab stehen.
2. Benennen Sie in Ihrem Projekt die main.c um in echo\_server.c.
3. Kopieren Sie den Quellcode des Echo-Servers in die echo\_server.c. Sie können den Quellcode hier herunterladen:  
[https://git.fh-muenster.de/schinzel/PSE2017-Test/blob/master/echo\\_server.c](https://git.fh-muenster.de/schinzel/PSE2017-Test/blob/master/echo_server.c)
4. Committen und Pushen Sie Ihre Änderungen ins Repository. Wählen Sie dafür „VCS“ → „Commit Changes“. Schreiben Sie eine aussagekräftige „Commit Message“ und wählen Sie „Commit and Push“.
5. Überprüfen Sie, dass die Daten erfolgreich ins Gitlab übertragen wurden.

## Den Echo-Server ausführen

1. Starten Sie den Server, indem Sie auf den grünen „Run“-Knopf klicken.



## Den Echo-Server testen

1. Öffnen Sie ein Terminal und geben Sie den Befehl `nc localhost 31337` an.
2. Geben Sie jetzt etwas Text ein und bestätigen Sie.
3. Der Server sollte den von Ihnen eingegebenen Text empfangen und wieder auf der Kommandozeile ausgeben.

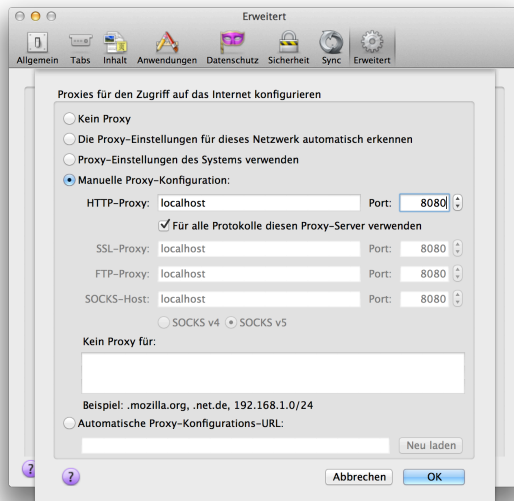
## Testen Sie den Echo-Server jetzt mit einem Web-Browser

1. Starten Sie Firefox und besuchen Sie die URL `http://localhost:31337/`
2. Der Browser zeigt jetzt etwas Text an.
3. Was bedeutet dieser Text?

## Installieren Sie den HTTP-Proxy

1. Laden Sie die burp-Suite von der URL <https://portswigger.net/burp/freedownload> herunter. (Auf den Laborrechnern liegt sie bereits unter programme/.)
2. Starten Sie den Burp-Proxy von der Kommandozeile über den Aufruf  
`$ java -jar burpsuite_...`

Machen Sie sich vertraut mit der Konfiguration von Firefox und richten Sie ein, dass der burp-Proxy für alle Anfragen verwendet wird („Preferences“ → „Advanced“ → „Network“ → „Settings“). Die Konfiguration sollte wie folgt aussehen:



3. In burp, wechseln Sie auf dem Reiter „Proxy“ und schalten Sie „Intercept is on“ auf „Intercept is off“
4. Stellen Sie jetzt in Firefox einige Anfragen an den Echo-Server.
5. Wechseln Sie in burp auf den Reiter „HTTP History“ und machen Sie sich mit den HTTP-Anfragen („Requests“) und die Antworten („Responses“) vertraut.
6. Wechseln Sie jetzt wieder in den Modus „Intercept is on“ und setzen Sie im Browser wieder eine Anfrage ab.
7. Ändern Sie den User-Agent in der HTTP-Anfrage von „Firefox“ auf „FH-Muenster-Browser 1.0“ und drücken Sie auf „Forward“.
8. Zeigen Sie, dass am Echo-Server tatsächlich der veränderte User-Agent ankam.

## **Ändern von Werten innerhalb des Debuggers**

1. Stoppen Sie die laufende Instanz des EchoServers, indem Sie im Clion auf das kleine rote Quadrat klicken.
2. Setzen Sie einen Breakpoint in Zeile 121, indem Sie rechts neben die Zeilennummer klicken. Ein roter Punkt erscheint jetzt rechts neben der Zeilennummer.
3. Starten Sie den Server im Debugger, indem Sie auf das kleine „Käfer“-Icon klicken und starten Sie die Ausführung mit einem Klick auf den grünen Pfeil („Resume Program“).
4. Stellen Sie jetzt in Firefox eine Anfrage an den Echo-Server.
5. Im „Variables“-Fenster ändern Sie den Inhalt der Variablen `buffer` auf einen anderen Wert, z.B. `(char*) "Hallo Welt!"`.
6. Zeigen Sie, dass der Echo-Server tatsächlich den veränderten Wert zurückgegeben hat.

## **Abnahme am Ende Ihres ersten Praktikum-Termins**

1. Legen Sie den Projektplan Ihrer Gruppe vor.
2. Zeigen Sie, dass Sie die Anfragen im burp-Proxy ändern können.
3. Zeigen Sie, dass Sie die Antworten im Clion-Debugger ändern können.

## **Einführung in Gitlab!**