**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare Informatică și Microelectronică**

**Departamentul** **Ingineria Software şi Automatică**

**Raport**

**Lucrare de laborator nr.3**

Disciplina: Programarea în rețea

Tema: HTTP Client

A îndeplinit: st.gr.TI-172 Serețan Evelina

A controlat: Buldumac Oleg

Chișinău 2020

**Scopul lucrării:**

Să se creeze o aplicație client HTTP

**Răspunsuri la întrebări:**

1. Cum este formatat corpul unei cereri HTTP pentru o cerere HTTP de tip POST?

În calcul, POST este o metodă de solicitare acceptată de HTTP folosită de World Wide Web. Prin proiectare, metoda de solicitare POST solicită ca un server web să accepte datele incluse în corpul mesajului de solicitare, cel mai probabil pentru stocarea acesteia. Este adesea utilizat la încărcarea unui fișier sau la trimiterea unui formular web completat..În cererile POST, valorile sunt trimise în „corpul” cererii. Cu formularele web, acestea sunt cel mai probabil trimise cu un tip de aplicație / x-www-form-urlencoded sau multipart / form-data.

1. De unde știe un client HTTP ce tip de conținut trimite serverul HTTP?

Tipul de conținut îl arată antetul Content-Type,care indică răspunsul de la cerere.

1. Cum decide un client dacă ar trebui să aibă încredere în certificatul unui server?

Clientul descarcă certificatul unui server care conține cheia publică a serverului. Clientul utilizează o cheie publică pentru a verifica dacă întradevăr cerficatul este emis de o autoritate de certificare. Certificatul conține adresa IP a serverului, iar clientul verifică dacă IP – urile coincid. Dacă IP- urile coincid și dacă certificatul încă nu a expirat, atunci clientul are încredere în certificatul respectiv.

1. Care este problema principală cu certificatele autosemnate?

Certificatele autosemnate nu asigură securitate, ș datele transferate devin vulnerabile.

1. Conexiunea persistentă HTTP – care sunt principalele beneficii?

Beneficiul este că nu e neapărat să formăm handshake- ul de fiecare dată cînd se trimite o cerere, iar asta duce la câștig în timp și nu duce la supraîncărcarea rețelei.

1. Ce este negocierea conținutului în HTTP și cum are loc?

În HTTP, negocierea conținutului este mecanismul care este utilizat pentru difuzarea diferitelor reprezentări ale unei resurse la același URI, astfel încât agentul utilizator poate specifica care este cel mai potrivit pentru utilizator (de exemplu, ce limbă a unui document, ce format de imagine sau care conține codificarea).

1. Care sunt tipurile de negociere a conținutului HTTP și cum au loc?

Există două tipuri de negociere a conținutului: negociere proactivă și negociere reactivă. Negocierea conținutului bazată pe server sau proactivă se realizează prin algoritmi de pe server care aleg între reprezentările posibile ale variantei. Aceasta se realizează în mod obișnuit pe baza criteriilor de acceptare furnizate de către utilizatori. Negocierea conținutului bazat pe agent sau reactiv este realizată de algoritmi din agentul utilizator care aleg între reprezentările posibile ale variantei. Aceasta se realizează în mod obișnuit pe baza listei de reprezentări și metadate ale serverului furnizate de server.

1. Ce este un ETag în HTTP și cum funcționează?

Etagul sau eticheta entității face parte din HTTP, protocolul pentru World Wide Web. Este unul dintre mai multe mecanisme pe care le oferă HTTP pentru validarea cache-ului Web, care permite unui client să facă cereri condiționate. Acest lucru permite cașelor să fie mai eficiente și economisește lățimea de bandă, deoarece un server Web nu trebuie să trimită un răspuns complet dacă conținutul nu s-a schimbat. ETag-urile pot fi, de asemenea, utilizate pentru controlul optim al concurentei, ca o modalitate de a ajuta la prevenirea actualizărilor simultane ale unei resurse de a se suprascrie reciproc.

1. Diferența dintre protocoalele fără stare și cele cu stare. Cărui tip îi aparține HTTP?

Protocoalele cu stare mențin o conexiune între server și client până când transferul de date s-a finisat, pe când protocoalele fără stare nu mențin nici o conexiune între client și server. HTTP este un protocol fără stare, dar poate simula starea prin cookies, sau sesiune.

1. Avantajele cheie ale HTTP/2 în comparație cu HTTP/1.1

* Utilizarea multiplexării
* Introducerea funcționalității server-push
* Arhivarea anteților utilizând HPACK
* Protocol binary
* Push Server HTTP / 2
* Compresia anteturilor cererii

1. Ce este un tip MIME, din ce constă și pentru ce se folosește?

**Multipurpose Internet Mail Extensions** (**MIME**) este un standard de Internet care extinde formatul de [e-mail](https://ro.wikipedia.org/wiki/E-mail) pentru suportul:

* Textului în altele seturi de caractere decât [ASCII](https://ro.wikipedia.org/wiki/ASCII)
* A atașamentelor non-text: audio, video, imagini, programe etc.
* A corpurilor de mesaje cu mai multe părți
* A informațiilor de antet (header) în seturi de caractere non-ASCII.

1. Care este diferența dintre GET și POST?

GET este o metodă folosită pentru cereri de extragere a resurselor de pe server, POST pentru transmiterea datelor pe server.

1. Care este diferența dintre PUT și POST?

PUT se folosește pentru update, POST pentru a crea noi entități.

1. Care sunt metodele idempotente în HTTP?

HEAD,PUT, DELETE, GET, OPTIONS.

1. Care sunt metodele sigure și nesigure în HTTP?

Sigure: OPTIONS, GET, HEAD.

Nesigure: PUT, POST, DELETE, PATCH

1. Pentru ce este nevoie de cURL?

Pentru a automatiza transferul de date dintre client și server.

1. Pentru ce este nevoie de HTTP Proxy?

Pentru securitate, controlul accesului la internet, anonimitate, viteza sporita a internetului.

1. Diferența dintre autentificare și autorizare

Autentificarea și autorizarea sunt două mecanisme utilizate în aceste sisteme pentru a asigura informații. Autentificarea este utilizată pentru a identifica un anumit utilizator pentru ai permite accesul la un sistem.  După autentificarea utilizatorului în sistem, autorizarea oferă limitele necesare și accesează utilizatorul. Aceste politici sunt definite într-un firewall sau într-o listă de control al accesului pe un server de fișiere. Autorizarea se aplică numai utilizatorilor autentificați.

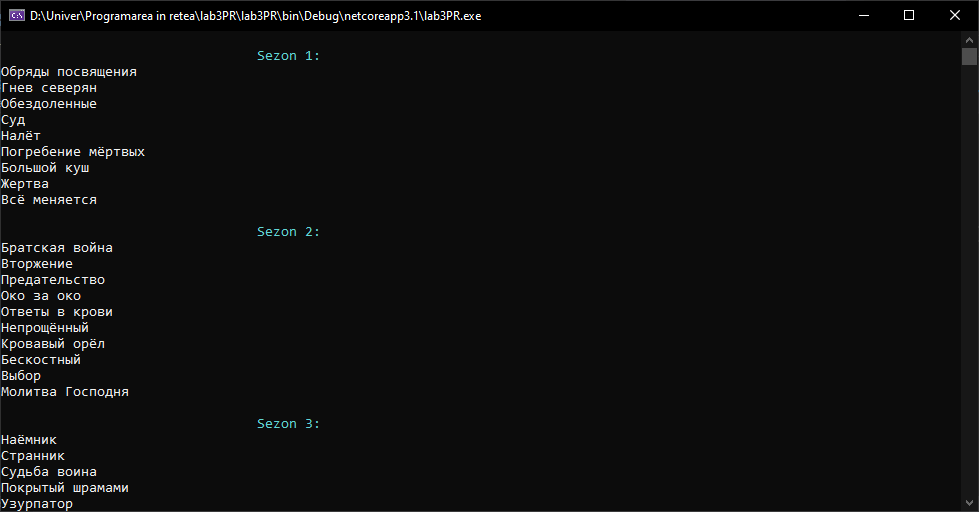
1. Care sunt metodele de autentificare HTTP?

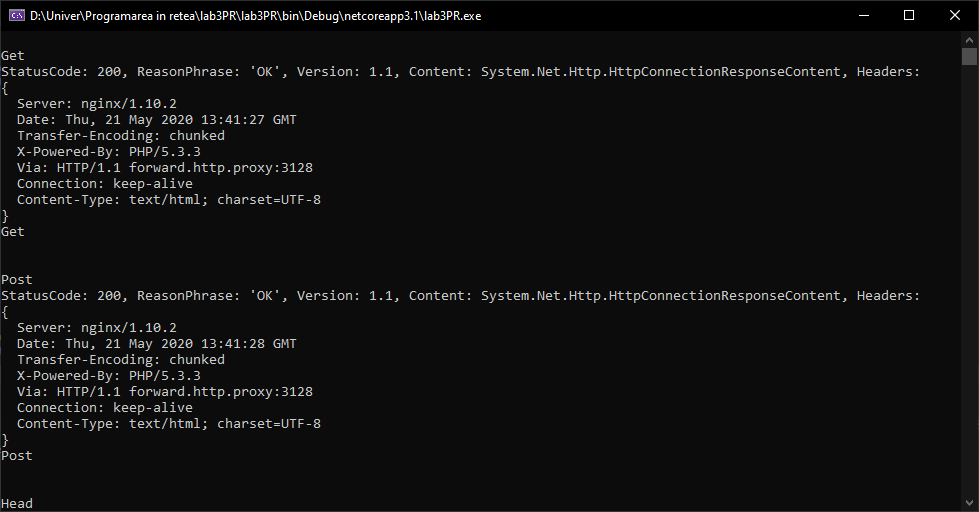
Prin Semnături, Tokens, Cookies, parolă pentru unică folosință, Basic HTTP.

1. HTTP cookies – pentru ce se folosește?

Un cookie [HTTP](https://ro.wikipedia.org/wiki/HTTP) sau un modul cookie este un text special, deseori codificat, trimis de un [server](https://ro.wikipedia.org/wiki/Server) unui [navigator](https://ro.wikipedia.org/wiki/Browser) [web](https://ro.wikipedia.org/wiki/World_Wide_Web) și apoi trimis înapoi (nemodificat) de către navigator, de fiecare dată când accesează acel server. Cookie-urile sunt folosite pentru [autentificare](https://ro.wikipedia.org/wiki/Autentificare) precum și pentru urmărirea comportamentului utilizatorilor; aplicații tipice sunt reținerea preferințelor utilizatorilor și implementarea sistemului de „coș de cumpărături”.

Rezultate obținute:





Concluzie:

In urma efectuării acestei lucrări de laborator s-au acumulat competențe de lucru cu protocolul HTTP. De asemenea au fost căpătate cunoștințe în ceea ce privește conectarea la proxy server. Tot prin proxy au fost evectuate și request-urile la pagina web aleasă. Au fost folosite următoarele metode ale protocolului HTTP ca GET, POST, HEAD, OPTIONS.

Cu ajutorul expresiilor regulate, au fost extrase echipele din clasamentul campionatului intern a Spaniei. În final au fost analizate rezultatele și formulate concluzii.

**Github Link:** <https://github.com/Seretaneva/PR>