

CSc 4220/6220 – Fall 2018

Assignment #2 – Application Layer

Deadline: Tuesday, September 18th 11:59 pm

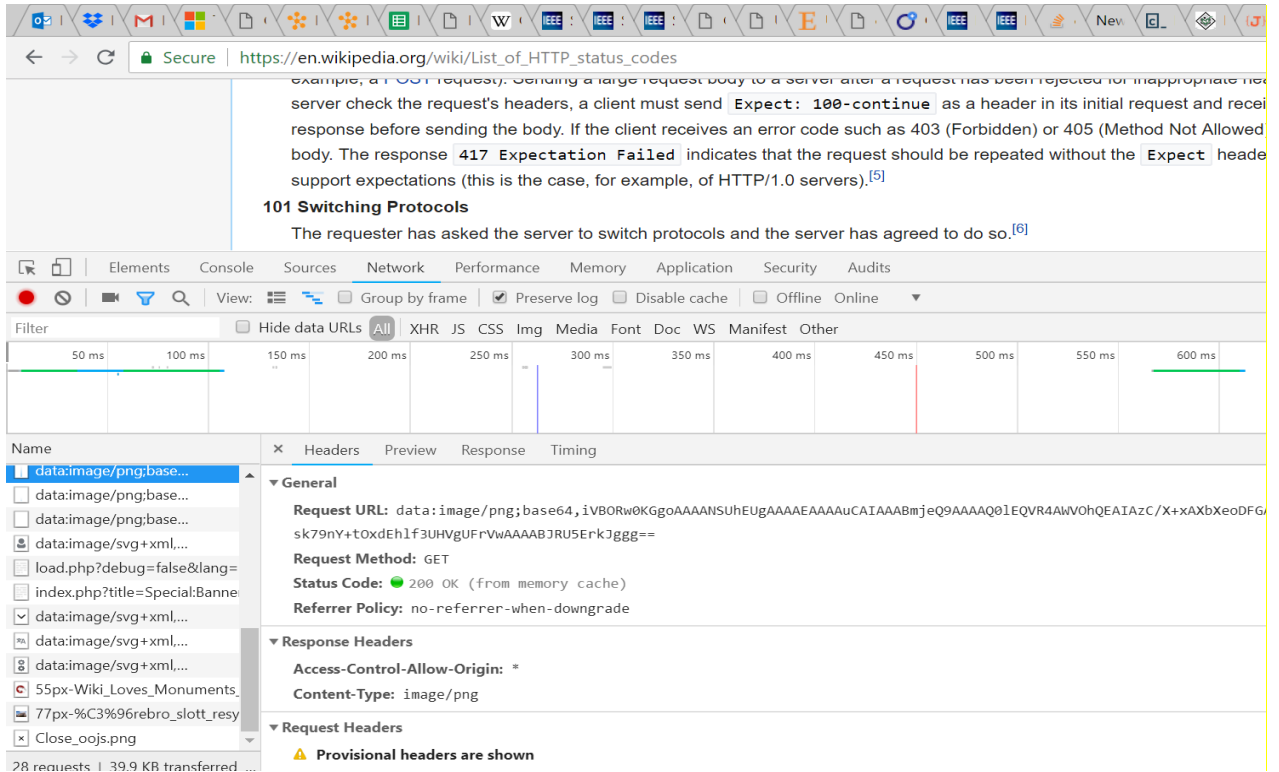
Late Deadline: Saturday, September 22nd 11:59 pm

Directions: Use Java/C++/Python to program your solutions to this Assignment. Answer all questions. Upload your assignment programs in a single document along with their outputs with screenshots on their execution. Make sure you document your code.

1. Write a server program using TCP that accepts requests from clients to capitalize strings sent by a client. When a client connects, a new thread is started to handle an interactive dialog in which the client sends a string and the server sends back the capitalized version of the string. Set a maximum limit of 10 strings to accept from a client. The server capitalizes each string and sends it back to the client with a prompt for the next string. You may choose to terminate the connection before the maximum limit of 10 strings is reached. Send an appropriate message to the server in that case when it prompts for the next string. After the maximum limit of 10 strings is reached, your program should close the connection automatically and display the message "Connection Closed." and terminate execution. You should write the corresponding client program as well. Test the execution of the program by running with different cases and providing screenshots of the output.
2. Write a client-server program to work like an ATM machine using UDP protocol. The client program should have the ability to choose operations, such as deposit, withdraw and balance check of the user if he is a valid user. And server program maintains a file with user details for authentication. For each row we would have user details including name, pin, and available balance. When a client requests to deposit or withdraw money, then the server should prompt them for their id and pin and after their validation, the server should allow them to deposit or withdraw money (On deposit add the money to the available balance and on withdraw deduct the available balance by the amount given and save this information to the file, and also provide some sentence to client program that the operation is successfully done). Test the operations from client program like deposit + balance check, withdraw + balance check, invalid user trying to withdraw/deposit/checking balance (show an error message that he is not a user) and provide screenshots of their output.

NOTE: Maintain a txt file with predefined user details with a space/tab separating their details or you can add one more operation that will create a new account with the details as above also.

3. Request any webpage from your system and try to observe all types of status codes (2**, 3**, 4** and 5**) in the http responses for their respective http requests. You can do this via Wireshark or using DEVTOOLS in your browser ([Press F12 or Ctrl + Shift + I] and see networks tab which has series of http requests within that webpage, see the screenshot below using DevTools)



Graduate Students:

Provide a multi-threaded server program which has same functionality as of 1st problem and it can connect to multiple clients and communicate with them synchronously using TCP protocol.