Q1)

Although this basic proxy server acts as a basic filter between the user and a web server, it has many significant limitations.

The most noticeable immediately is that it can only be used for http requests rather than https, which majority of modern websites use. This means that SSL/TLS cannot be used with this proxy server. HTTPS is essential for sensitive data such as payment details so that the origin of the server can be verified as genuine and guarantees some level of encryption. It is also only configured to use http 1.1 and cannot perform GET requests for purposes such as data forms.

It is also significantly limited in the way it handles subsequent requests for the same website. For websites with images or other content that needs to be downloaded, these requests do not go through the proxy server. Additionally, many browsers request a thumbnail image to display in the website tab. In Chrome and Firefox this is ‘favicon.ico’. When this request is sent, it is expected that the context of the previous hostname is conserved. However, our proxy server handles each request separately and is effectively attempting to request from a hostname called ‘favicon.ico’.

Another limitation is that redirects from one website to another are not handled through the proxy server, but via the browser directly. An issue with this is that if a whitelisted website contains a link to a blocked website, it will still redirect the user and bypass the proxy server. This is a risk for websites that have user content such as public forums where links could be malicious. Any javascript returned will also be executed by the browser, not handled by the proxy server.

The proxy server can only handle one client at a time. This means that if multiple browsers (or users) are attempting to use the proxy server at once, they will get slow or potentially no service. It has been noted that browsers send a TCP keepalive packet to the proxy server, keeping it waiting for more data. If another client attempts to connect, the proxy server will be blocked meaning that this new user’s requests will not be served.

In its current form, the proxy server is not actually performing any blocking for certain websites. It is simply acting as a passthrough

Q2)

Https could be supported by using X.509 certificates and the python ssl package that supports TLS. This is usually implemented by the client authenticating the server. Before connecting, the socket is wrapped. The web server then sends a certificate, and the client checks the certificate chain to see if it is signed by a trusted CA before sending data. This also adds encryption to any data sent between the proxy server and the web server.

Q3)

The proxy server uses TCP sockets. This is controlled when the socket is created, by setting the socket type to SOCK\_STREAM. This has been confirmed by observing Wireshark output indicating that TCP is used.

DNS, IP, HTTP, Ethernet II are the other protocols used by the Proxy server. DNS is used to translate the hostname into an IP address. The IP header is used to specify the source and destination IP addresses and ports between the browser (client), proxy server and web server. Ethernet II is used to allow the MAC addresses of the local router and proxy server to be specified. This allows communication between the proxy server and the requested web server. HTTP is used to dictate the format of the request between the client, proxy and web server.

Q4)

Powering on. This is not a protocol as such, but this allows the NIC card to send and receive network traffic.

Now the laptop needs to be assigned an IP address so it can be identified on the local network. This is carried out using DHCP. Initially the laptop sends a DHCP discover message to the broadcast IP address of 255.255.255.255, enabling all hosts to receive this. DHCP servers are configured to respond. In a typical home network this is the router. It responds with a DHCP offer containing fields such as an IP address, subnet mask and DNS servers. It is possible for multiple DHCP servers to respond and offer different IP addresses. It is up to the laptop’s operating system to choose one of these.

ARP to get IP address for gateway router. The laptop sends out an ARP request to the broadcast MAC address of FF:FF:FF:FF:FF:FF. Since it knows the gateway router’s IP address from DHCP and has an IP address itself, it sends a whois <gateway IP> tell <laptop IP>. This is propagated to all hosts on the subnet. This request uses the Ethernet header to specify its own source MAC address and routers are configured to reply to this request with a sender MAC address and sender IP address. A protocol is specified in the ARP message, this is commonly IPv4.

Now that the IP and MAC address of the gateway router is known, DNS is used to translate the hostname provided in the URL to an IP address. As there is no local DNS data cached by the browser, the laptop sends a DNS request to a known DNS server (typically the local router) on port 53. This DNS server checks any DNS records it may have. If it does not have a record it replies to the laptop with a further list of DNS servers to try and send a request to. This continues until the relevant DNS server is reached that can reply with the IP address of the web server. DNS can use either UDP or TCP, however UDP is the default. Caching is performed at each DNS server and the browser so that subsequent requests can be sped up.

Since the IP address of the web server is known, the laptop establishes a TCP connection with the web server. It does this through the TCP three way handshake. The browser sends a TCP SYN packet to port 80 (HTTP) of the web server. It responds with a SYN-ACK and finally the browser sends an ACK packet to complete the connection. Often PSH ACK is used, meaning that data is sent immediately in the same packet when completing the handshake. In this case, this packet will contain HTTP data. TCP allows for reliable communication between the browser and web server. This allows lost packets to be re-sent, congestion window size to be calculated to increase throughput and packets to be reassembled in order.

Over this established TCP connection, the browser sends a HTTP GET request corresponding to the requested webpage.