1)

UDP is really useful for sending packet without first establishing a connection to a recipient. We then accentuate the speed of the sending package with the informations we need after, the filename, the hostname and port. We could loose some informations while the UDP packet tries to reach the recipient (the server). UDP provides no guarantee for delivery and no protection from duplication, but the simplicity of UDP reduces overhead from the protocol. We can read the packet when they come to be sure it has no lack of information, or just send multiple packets that will arrive at some point. (not sure about that)

2)

A stream socket (TCP) is better than a datagram socket (UDP) for a file transfer because a file could be heavy, and the heavier the infos are the more chance you have to loose informations with UDP. TCP has a built-in feedback mechanism checks and acknowledges whether the data was received correctly. If any data is missing or lost, further mechanisms retransmit the corrupt or missing information. These mechanisms make TCP the perfect protocol for transferring information such as still images, data files (like spreadsheets and documents) and web pages (including graphics, text and the languages that interpret how things are meant to display). We would have to implement the PrintWriter, BufferReader and all the functions manipulating files.

3 )

The server could multiple clients in parallel by using multi thread, doing so each thread would correspond to a client and could exchange files.