

## Projeto 2 IARC -Protocolo de transferência de ficheiros com cache

### Relatório

Foi lançado um segundo projeto na cadeira de Introdução às Redes de Comunicação(IARC), em que nos foi pedido que criássemos um protocolo de transferência de ficheiros entre um cliente e um servidor, que permita ao cliente listar os ficheiros do servidor, efectuar operações de 'download' (transferência de ficheiros do servidor para o cliente) e do oposto isto é, fazer 'upload' (transferência de ficheiros do cliente para o servidor), tendo em conta que o cliente pode ser um utilizador registado, ou um convidado.

Começamos por apresentar um pequeno menu que nos permite não só distinguir um cliente de um convidado, como registar um utilizador. Criamos também um 'socket', que nos permite a comunicação do cliente com o servidor, no caso de ele estar registado. Caso ele seja convidado criamos dois 'sockets' um deles permite a ligação 'cliente-cache' e o outro 'cache-servidor'. A 'cache' é uma ponte entre o servidor e o cliente, que permite o armazenamento de ficheiros localmente, impedindo assim que o servidor seja sobrecarregado.

Aquando do registo de um utilizador, o cliente vai comunicar diretamente com o servidor e este vai verificar se o utilizador já foi criado. Em caso afirmativo, devolve uma mensagem ao utilizador, que deverá fazer novamente o registo, caso contrário o utilizador é aceite, podendo de seguida fazer o 'login'.

Quando um utilizador entra na sua conta, o socket vai enviar uma mensagem para o servidor, que vai verificar se o nome de utilizador existe e se a palavra chave introduzida corresponde ao 'username'. Caso contrário, o servidor vai continuar a fornecer acesso do primeiro menu ao utilizador

Assim que o utilizador entrar no servidor, quer através de um 'user' registado, ou de um convidado, ser-lhe-à fornecido um menu principal, que possibilita o envio/receção de ficheiros, assim como a listagem dos mesmos, continuando neste menu até que o utilizador insira o comando 'LOGOUT'.

Utilizamos um código modelar, para facilitar o acesso a cada comando inserido pelo utilizador e para facilitar a identificação de funções.

Temos também no ficheiro do servidor, na função 'main' uma condição que verifica a existência do ficheiro que possui a lista de utilizadores. Caso este ficheiro exista, eles são carregados para um dicionário ('users'), que possui os utilizadores e as palavras chave, caso contrário vai criar este ficheiro.

As funções que são chamadas aquando do uso de um comando por parte de um utilizador (excetuando o 'LOGOUT') foram criadas com base nos conhecimentos obtidos à cerca de redes ao longo do semestre e que nos permitem a comunicação entre dois aplicativos, através de uma rede.

Escolhemos a linguagem de Python porque ao contrário de C não trabalha tão próximo do processador, sendo também uma linguagem de alto nível, que permite um fácil acesso aos objetos criados, tal como Java. Não foi escrito em Java porque, apesar do acesso aos objetos também ser fácil, o nosso conhecimento nesta linguagem podia não permitir a elaboração deste projeto.

Trabalho realizado por:  
Daniel Azevedo nº2014200607  
Nuno Afonso Santos nº201426541