



Sistemas Operativos
2022/2023

Relatório de Trabalho Prático - Meta 2

Pedro Emanuel Dinis Serrano - 2016017926

Makefile

Regra	Resultado
all	Compila todos os programas
env	Compila o programa env
frontend	Compila o programa frontend
backend	Compila o programa backend
clean	Remove os ficheiros executáveis

Geral

Header file que contém todas as structs que são utilizadas ao longo do trabalho, assim como os todos os *defines*.

As estruturas presentes neste ficheiro são:

- Item - contém toda a informação necessária para cada item;
- Promotor - contém a informação que cada promotor envia, assim como o seu nome e *pid*;
- Msg - estrutura usada para enviar mensagens do *backend* para o *frontend*;
- User - contém as informações de cada *user*, assim como o seu *heartbeat* e, caso esteja a enviar um comando, o tipo de comando e os argumentos do mesmo;
- Node - estrutura principal do *backend* que contém os *arrays* de utilizadores, itens e promotores. Contém, também, o número de utilizadores, itens e promotores ativos. para além disto, é onde está inserido o trinco das *threads*, o *id* total para ser aplicado a novos itens, o *heartbeat* definido pelas variáveis de ambiente para poder dar *reset* a um utilizador quando necessário e o nome do ficheiro dos utilizadores.

Env

Contém as funções necessárias para ler o valor das variáveis de ambiente que contém o nome dos ficheiros relativos aos utilizadores, itens e promotores e *heartbeat*.

Backend

O *backend* lê e valida os comandos do administrador, assim como os argumentos de cada um.

O *backend* contém 2 *named pipes*: *FIFOCOMANDOS*, para onde todos os utilizadores enviam os seus comandos e de onde recebem as respostas ao mesmos, e *FIFOSERVER*, para onde todos os pedidos de *login* são enviados e devidamente respondidos.

No *backend* existe 3 threads:

- *threadClientes* - thread que recebe e executa os comandos de qualquer cliente ligado à plataforma;
- *t[0]* (*recebeClientes*) - thread que está constantemente à espera de novos pedidos de *login*.
- *t[1]* (*sinalVida*) - thread que verifica se os utilizadores devem ou não ser expulsos devido à inatividade;
- *t[2]* (*tempoItems*) - thread que verifica se o tempo de leilão do item já chegou ao fim ou não.
- *threadPromotores* - threads que são lançadas para que cada uma corra um promotor.

Frontend

O *fronted* é responsável por validar os comandos introduzidos pelo utilizador e, posteriormente, enviar os mesmo para o backend. Apenas existe um *named pipe* por cada *frontend*.

Decidi implementar um *select* no cliente uma vez que este apenas tem duas possíveis entradas, o *stdin* e o seu *named pipe*, pelo que uma parte do *select* está “à escuta” dos comandos introduzidos pelo utilizador, e a outra à escuta no *named pipe* do cliente para qualquer mensagem que possa chegar da parte do *backend*.

Conclusão

Este trabalho trouxe diversos desafios para os quais eu não estava à espera, pelo que foi uma boa maneira de aprender e desenvolver o meu conhecimento. Apesar de não ter conseguido resolver alguns deles, principalmente a parte dos promotores, sinto que aprendi bastante, uma vez que o resto dos desafios consegui enfrentar com sucesso.