



Departamento de Engenharia Informática e de Sistemas

Instituto Superior de Engenharia de Coimbra

Licenciatura em Engenharia Informática

Sistemas Operativos 2021/2022

Relatório Técnico

MEDICALso

Diogo Pascoal 2018019825

Nuno Honório 2019126457

Índice

1. Introdução.....	3
2. Estrutura do projeto.....	4
2.1. Arquitetura.....	4
2.1.1. Estruturas	4
2.1.2. Listas ligadas.....	5
2.1.3. Threads.....	6
2.1.4. Selects	7
3. MakeFile.....	8
4. Conclusão	8

1. Introdução

Este trabalho prático consiste na implementação de um sistema para gerir o atendimento de clientes em estabelecimentos médicos. O sistema, denominado MEDICALso, destina-se a correr em ambiente Unix linha de comandos e irá mediar a interação entre doente, médico e balcão de atendimento.

O mesmo foi realizado em C e foram utilizados os conceitos aprendidos durante a unidade curricular.

Neste relatório estará descrita a arquitetura adotada e os métodos implementados para concretizar o pedido.

2. Estrutura do projeto

2.1. Arquitetura

Para a realização deste projeto utilizamos uma arquitetura cliente servidor, onde o servidor é o balcão e os clientes são os utentes e os especialistas. Além disso, o balcão está constantemente em ligação com o classificador de modo a obter a classificação atribuída ao utente.

Cada utente e cada especialista usa um terminal diferente, que o permite interagir com o balcão ou outros clientes.

O servidor, neste caso o balcão, é responsável pela gestão geral do sistema, desde a conexão do utente ao especialista atribuído à listagem dos mesmos. Caso este termine todo o sistema deve terminar.

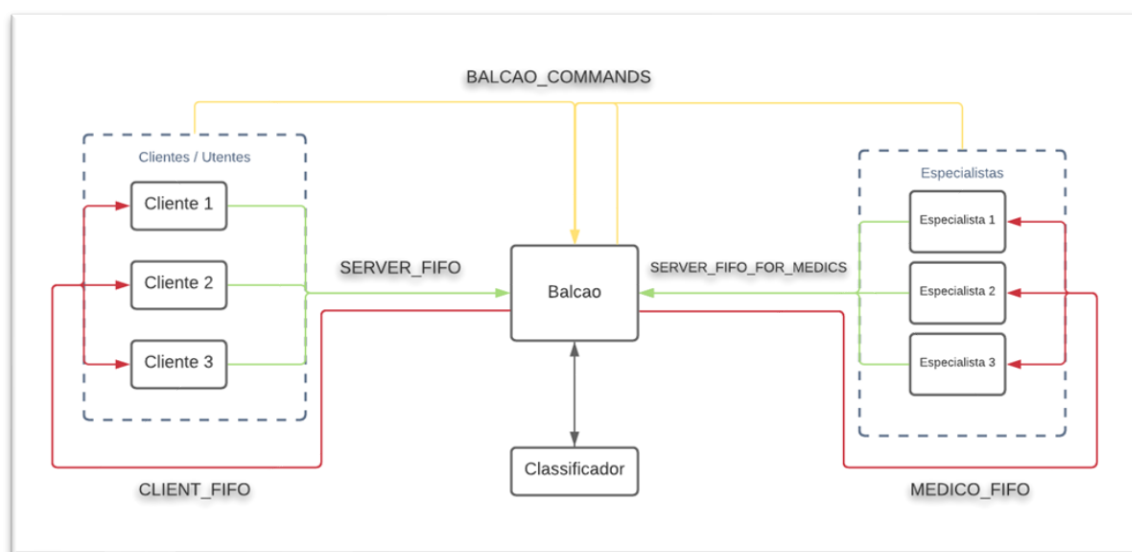


Figura 1 - Arquitetura de Comunicação (Pipes)

2.1.1. Estruturas

Balcao.h

O header *balcao.h* tem como objetivo definir estruturas necessárias à gestão geral do sistema. Dentro deste header, encontramos a struct *Balcao*, que contem o número máximo de utentes e especialistas, assim como o número destes ligados de momento. Também conta com um array que contém o número de utentes em espera para cada especialidade: oftalmologia, neurologia, estomatologia, ortopedia e geral, respetivamente.

Encontram-se também struct que guardam toda a informação necessária para as diversas threads funcionarem.

Cliente.h

No header cliente.h está declarada a estrutura *Utente*, contendo todos os dados dos utentes, tais como nome, o seu pid, sintomas, especialidade atribuída, prioridade atribuída... Além disso, contém um ponteiro para um possível próximo utente na lista ligada, visto que os utentes estão guardados numa lista ligada. Também lhe é fornecido o nome do médico que lhe foi atribuído.

Medico.h

A estrutura *especialista* é idêntica à *cliente*, contendo as informações sobre cada médico, tais como pid, nome, especialidade.... Além disso, contém também o ponteiro para o próximo especialista na lista, o pid do balcão, o nome do utente que lhe foi atribuído e o pid do mesmo.

Utils.h

Por fim, no header *utils.h*, para além de incluir as diversas bibliotecas que são necessários em todos os programas deste sistema, temos a estrutura *MSG* usada para armazenar as mensagens enviadas entre os utentes, médicos e o balcão.

Também é neste header que se encontra o template para o caminho dos FIFO's usados pelo sistema.

2.1.2. Listas ligadas

Para armazenar os utentes e os especialistas foram criadas duas listas ligadas.

Sempre que é adicionado um especialista o mesmo é adicionado ao final da respetiva lista. Já para os utentes, estes são adicionados segundo a prioridade que lhes foi atribuído ao entrar no sistema.

Quando um utente ou especialista é removido da lista, é libertada a memória e é considerado que o especialista ou o utente abandonou o sistema.

2.1.3. Threads

Foram desenvolvidas cinco threads para além da main thread no balcão.

Thread recebeUtentes

Esta thread é responsável por ficar à espera de atividade no pipe SERVER_FIFO, o que representa a tentativa de registo de um novo utente. Trata de todos os passos para o registo de um novo utente no sistema, assim como a notificação ao sistema e ao utente do seu registo.

Thread recebeMedicos

Esta thread é responsável por ficar à espera de atividade no pipe SERVER_FIFO_FOR_MEDICS, o que representa a tentativa de registo de um novo especialista. Trata de todos os passos para o registo de um novo especialista no sistema, assim como a notificação ao sistema e ao especialista do seu registo.

Thread apresentaStatusEspera

Esta thread é responsável por notificar o balcão do estado da lista de espera de utentes. Esta thread fica em pausa em cada ciclo 20 segundos inicialmente, sendo este valor ajustável pelos comandos do balcão.

Thread estabeleceContacto

Esta thread é responsável por emparelhar utentes e especialistas.

Thread handleHeartBeats

Esta thread é responsável por confirmar que os especialistas não pararam de dar sinal de vida. Caso um especialista falhe duas vezes o heartbeat, é considerado que este abandonou abruptamente o sistema sem notificar. A thread irá notificar o balcão e o utente (caso o especialista estivesse em consulta) e procede à remoção do especialista na lista, assim como o término da consulta do utente.

Quando é indicado ao balcão para encerrar, todas as threads terminam o seu trabalho e é terminada a execução do balcão.



O programa do especialista também tem uma thread, heartbeat, que se encarrega de enviar ao balcão o sinal de vida.

2.1.4. Selects

Foram desenvolvidos 3 selects, um em cada programa do sistema.

Balcao

O balcão tem um select que fica à espera de input do teclado ou do pipe BALCAO_COMMANDS. Este pipe serve para os clientes notificarem o balcão de diversos acontecimentos (utente desistiu, especialista terminou consulta, especialista vai encerrar, etc.). Deste modo o balcão aceita comandos escritos no teclado ou enviados para o mesmo através de pipes.

Cliente / Utente

O cliente tem um select que fica à espera de input do teclado ou do seu pipe CLIENT_FIFO, sendo que o input do teclado são comandos para indicar que vai terminar a consulta ou abandonar, ou texto que é enviado para o especialista se estiver conectado a um. Ao mesmo tempo pode receber texto ou ser notificado pelo balcão de alguma instrução, sendo esses eventos processados aquando do uso do pipe CLIENT_FIFO.

Especialista

O especialista tem um select que fica à espera de input do teclado ou do seu pipe MEDICO_FIFO, sendo que o input do teclado são comandos para indicar que vai terminar a consulta ou abandonar, ou texto que é enviado para o cliente se estiver conectado a um. Ao mesmo tempo pode receber texto ou ser notificado pelo balcão de alguma instrução, sendo esses eventos processados aquando do uso do pipe MEDICO_FIFO.

3. MakeFile

O nosso Makefile é composto de 5 regras:

- all : executa todos os targets
- balcao : cria o executável balcao
- cliente : cria o executável cliente
- medico: cria o executável medico
- clean : limpa qualquer ficheiro FIFO que possa não ter sido removido, assim como os executáveis

4. Conclusão

Após concluir a implementação do projeto e de testar o sistema não foi encontrado nenhum bug de execução que pusesse em causa a comunicação servidor cliente. Atendeu-se a todos os requisitos do enunciado.

Por fim, foi verificado e todas as threads terminam de forma ordenada e correta. Os “FIFOS” são removidos da pasta “/tmp/”.