Balcão

O balcão é onde a lógica toda do programa ocorre. Ele primeiro irá verificar a existência de duas variáveis de ambiente "**MAXCLIENTES**" e "**MAXMEDICOS**". Caso estas variáveis existem e sejam válidas ele ira correr o classificador para o ajudar na classificação de sintomas de possíveis clientes.

De seguida ele possui um select que fica a espera de receber comandos no teclado(caso o administrador queira executar alguma coisa) e fica também à escuta nos dois fifos que criou "**fifosvmed**" e "**fifosvcli**".

Caso ele receba uma mensagem do programa cliente ele lança uma thread que irá tratar de incorporar esse cliente no array de especialidades na lista ligada dos clientes dessa especialidade.

Caso receba uma mensagem do programa médico, ele irá executar uma outra thread que irá incorporar esse médico no array de especialidades na lista ligada dos médicos para aquela especialidade.

O balcão ira ter uma thread que irá ser ativada (através de uma variavel condicional) cada vez que uma consulta acaba ou um médico/ cliente é criado. Esta ira percorrer o array de especialidades e procurar se existem consultas, caso existam irá cria las e lançar uma thread que irá "vigiar" as consultas para saber quando acabam e lidar com isso.

Cliente

O programa é executado pela terminal utilizando o comando ./cliente <nome>
e pede ao utilizador os seus sintomas. Em seguida ele irá ser atendido pelo balcão e receberá a sua posição na fila onde aguardará. Caso a sua posição mude, ele irá receber a sua nova posição.A qualquer momento ele pode desistir ao escrever "**desisti**"

Se houver um médico disponível para o atender ele irá receber um aviso a dizer que está a ser atendido e poderá começar a dialogar com o médico. Ele pode acabar a consulta se escrever "**adeus**" e será removido do hospital.

Médico

O programa médico é semelhante ao programa cliente e pode ser executado utilizando o comando ./medico <nome> <especialidade>. Caso a especialidade não existe ele nunca irá receber clientes. Caso exista ele irá ficar em esperar para lhe ser atribuido um cliente.

Após lhe ser atribuido um cliente ele irá ser avisado e começa uma consulta da qual ele pode sair a qualquer momento ao digitar "**sair**" ou "**adeus**" se ele digitar sair, irá ser removido do sistema caso digite adeus este irá poder atender um novo cliente caso exista.

Comandos

O administrador pode digitar diversos comandos no balcão nomeadamente:

- utentes – demonstra o nome e pid de todos os utentes em espera / consulta e a especialidade onde se encontrão
- especialistas – demonstra o nome e pid de todos especialistas em espera / consulta e a sua especialidade
- delut X – apaga um utente com pid X
- delesp X – apaga um especialista com pid X
- freq N – Altera a frequência com que o balcão mostra um relatório dos utentes e especialistas em espera
- encerra – encerra o balcão e avisa todos os utentes e especialistas

Estruturas

Estas estruturas são utilizadas por todo o programa de modo, a proporcionar um fácil acesso a variáveis respetivas a cada cliente, médico etc. E a fornecer às threads variáveis às quais elas necessitam de acesso na suas respetivas funções

Cliente

Tipo	Nome	Objetivo
string	**Nome**	**Guardar o nome do utente**
string	**fifo_nome**	**Nome do fifo, especifico para cada cliente**
int	**PID**	**Guardar o id de processo do utente**
string	**Especialidade**	**Guardar qual o especialista necessário para este utente**
int	**Prioridade**	**Guardar a prioridade de atendimento do utente**
int	**posicao_lista**	**Posiçao na lista de espera**
int	**file_descriptor**	**Numero de file descriptor**
bool	**em_consulta**	**Guardar um valor de 0 ou 1 caso o utente esteja em consulta**
string	**especialidade**	**Especialidade associada ao cliente baseado nos sintomas**
Cliente*	**prox**	**Ponteiro para o próximo utente na lista ligada**
Medico*	**medico**	**Ponteiro para medico com que esta a ter a consulta, null se não estiver em consulta**

Medico

Tipo	Nome	Objetivo
string	**Nome**	**Guardar o nome do médico**
int	**PID**	**Guardar o id de processo do médico**
int	**fd**	**File descriptor do medico**
string	**especialidade**	**Guardar qual a especialidade do médico**
bool	**em_consulta**	**Guardar um valor de 0 ou 1 caso o médico esteja numa consulta**
string	**fifo_nome**	**Nome do fifo, especifico para cada medico**
medico*	**prox**	**Ponteiro para o próximo médico na lista ligada**

Especialidades

Tipo	Nome	Objetivo
int	**n_medicos**	**Número de médicos na lista**
int	**n_clientes**	**Número de clientes na lista**
char	**especialidade**	**Nome da especialidade**
struct Cliente*	**inicio**	**Inicio da lista ligada de clientes**
struct Medico*	**inicioMed**	**Inicio da lista ligada de médicos**

TDadosMedico

Tipo	Nome	Objetivo
int	**id**	**Id da thread**
int* fd	**fd**	**Ponteiro para o file descriptor**
medico*	**ptrmedico**	**Ponteiro para o médico que a thread vai utilizar**
struct Especialidades*	**listaespecialidades**	**Ponteiro para o array de estruturas do tipo especialidades**

Tconsulta

Tipo	Nome	Objetivo
Struct Especialidades*	**listaespecialidades**	**Ponteiro para o array de estruturas do tipo especialidades**
rcv_t*	**ptrRcv**	**Ponteiro a estrutura do tipo rcv_t**

Tchat

Tipo	Nome	Objetivo
medico*	**k**	**Ponteiro para o medico a dar a consulta**
cliente*	**j**	**Ponteiro para o cliente a receber a consulta**
rcv_t*	**ptrRcv**	**Ponteiro para estrutura do tipo rcv_t**
struct Especialidades*	**listaespecialidades**	**Ponteiro para o array de estruturas do tipo especialidades**

Ttimer

Tipo	Nome	Objetivo
struct Especialidades*	**listaespecialidades**	**Ponteiro para o array de estruturas do tipo especialidades**

Tsinal_vida

Tipo	Nome	Objetivo
struct Especialidades*	**listaespecialidades**	**Ponteiro para o array de estruturas do tipo especialidades**

send_t

Tipo	Nome	Objetivo
pid_t	**listaespecialidades**	**Ponteiro para o array de estruturas do tipo especialidades**
string	**nome**	**Nome do cliente que o balcão irá receber**
string	**msg**	**Mensagem envidada pelo cliente(sintomas) recebida pelo balcão**

rcv_t

Tipo	Nome	Objetivo
string	**fifo_name**	**Nome do fifo do cliente/medico**
string	**msg**	**Mensagem**

Trabalho realizado por:

- **Francisco Almeida – 2020138795**
- **Diogo Pinto - 2020133653**