



Licenciatura em Engenharia Informática

SISTEMAS OPERATIVOS

Manual de Utilizador

1. Introdução

Este manual descreve os procedimentos necessários para **compilar, configurar e executar** o servidor web desenvolvido no âmbito da unidade curricular **Sistemas Operativos**. O documento destina-se a utilizadores que pretendam executar e testar o servidor em ambiente Linux.

2. Requisitos do Sistema

Para utilizar o servidor web são necessários os seguintes requisitos:

- Sistema Operativo Linux
- Compilador **GCC**
- Suporte para **POSIX Threads (pthread)**
- Ferramentas opcionais para testes e validação:
 - [curl](#)
 - [ab](#) (ApacheBench)
 - [valgrind](#)

3. Compilação

Na diretoria raiz do projeto, executar o seguinte comando:

[make](#)

Este comando compila todo o código-fonte e gera o executável:

[webserver](#)

Para remover os ficheiros gerados durante a compilação, pode ser utilizado:

[make clean](#)

4. Configuração

O servidor é configurado através do ficheiro:

[server.conf](#)

Neste ficheiro podem ser definidos vários parâmetros, nomeadamente:

- Porta de escuta do servidor
- Número de threads de trabalho (workers)
- Tamanho da fila de espera de ligações
- Diretório base para os ficheiros estáticos

Os ficheiros a servir pelo servidor devem estar localizados no diretório:

[www/](#)

5. Execução do Servidor

Para iniciar o servidor web, executar:

```
./webserver server.conf  
make run
```

Após a execução, o servidor ficará a escutar na porta definida no ficheiro de configuração e pronto a responder a pedidos HTTP.

Para terminar o servidor de forma controlada, utilizar:

CTRL + C

6. Acesso ao Servidor

O servidor pode ser acedido através de um navegador web ou utilizando ferramentas de linha de comandos como o [curl](#).

Exemplo de pedido HTTP:

```
curl http://localhost:8080/index.html
```

O servidor suporta pedidos HTTP do tipo GET e pedidos com cabeçalhos adicionais, como pedidos [Range](#).

7. Execução dos Testes

Os testes automáticos do projeto encontram-se na diretoria:

tests/

Testes Funcionais

tests/test_functional.sh

Testes de Concorrência

make tests/test_concurrent
./tests/test_concurrent

Testes de Sincronização

tests/test_sync.sh

Testes de Stress

tests/test_stress.sh

Para ativar a verificação de fugas de memória com Valgrind:

`LEAKS=1 tests/test_stress.sh`

8. Encerramento e Limpeza

O servidor encerra corretamente ao receber um sinal de interrupção ([SIGINT](#)), libertando os recursos utilizados e garantindo que não permanecem processos zombie.

9. Conclusão

Este manual apresentou os passos necessários para a utilização correta do servidor web. O sistema encontra-se funcional, configurável e preparado para execução em ambientes Linux, cumprindo os requisitos definidos no enunciado do projeto.