



Sistemas Operativos

Docente: Pedro Rosa

Desenvolvimento de Shell para Linux

26/05/2024

Duarte Cavaleiro nº20211026

Francisco Santos nº20211206

Marco Camargo nº20211019

Mariana Gonçalves nº20210539

Índice

Problema e Solução	2
Arquitetura da Solução e tecnologias utilizadas.	3
Tecnologias a Utilizar	3
Resultados	4
Compile	4
Chatroom	4
Integração com AI	5
Game of Life	6
Autocorreção	6
Bibliografia	7

Problema e Solução

Nos últimos anos, temos vivenciado cada vez mais a integração de várias tecnologias avançadas no nosso dia-a-dia. Entre comunicação em redes e IA, é comum passar nosso tempo ao lado de tecnologias que seriam consideradas magia em qualquer outra época, de utilidade e facilidade de uso inigualável.

Isso compõe um grande problema a qualquer pessoa que comece a se interessar pelo desenvolvimento e pelos aspectos técnicos da informática, e não só em seu consumo. Enquanto essas tecnologias dominam o mundo, suas bases se mantêm da mesma forma há décadas, e a ausência das facilidades atuais compõem uma barreira de entrada imensa para possíveis novos engenheiros/as.

Nossa ideia é criar uma Shell para Linux que remedie essa discrepância ao dar o acesso ao usuário de várias das tecnologias que este já pode estar acostumado a utilizar. Ao integrar nossa shell com sistemas gráficos, conexão com a internet, IAs simples, entre outros, esperamos proporcionar uma experiência mais amigável, rápida e conveniente a usuários novos e experientes.

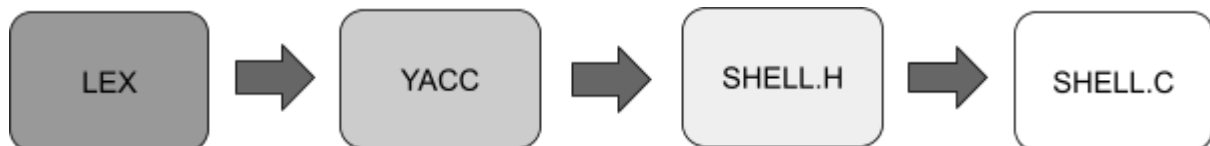
Em relação à proposta inicial fomos capazes de implementar:

- Chat rooms, sem a capacidade de partilha de ficheiros, que tinha sido proposto.
- Comando de compilar que automaticamente detecta a língua.
- Integração com a API de uma IA
- Visualização de gráficos básicos em shell
- Conway's Game of Life

Não conseguimos implementar a autocorreção, pois seria necessário alterar a arquitetura inteira do lex/yacc, No entanto todas as outras propostas foram implementadas levando em consideração vários casos irregulares de forma a criar soluções mais robustas.

Arquitetura da Solução e tecnologias utilizadas.

Para o desenvolvimento da shell, para a leitura e processamento do input do utilizador utilizamos o Lex e Yacc. Que por sua vez, ou corre diretamente o comando, usando “system()”, ou caso seja uma das funções implementadas pelo grupo roda a devida função.



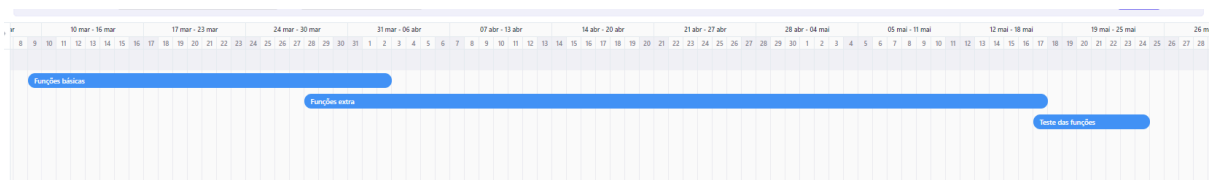
Para a implementação de chatrooms, é utilizado threads e sockets para comunicação entre utilizadores, e para a implementação de AI utiliza-se o Curl para se fazer pedidos HTTP, e o jansson para formatar o json o AI utilizado é o gemini da google através da API do Vertex AI.

Para duas funções, o Conway's Game of Life e a visualização de gráficos também é utilizado “ncurses” para o manuseamento mais fácil do terminal.

Tecnologias a Utilizar

- VScode/ C IDE
- GitHub
- Base de OS Linux
- Valgrind

Calendarização



Resultados

Compile

Consegue compilar, executar e listar os ficheiros que o utilizador quiser.

pontos fortes-

- Possível perguntar ao vertex ai do gemini que tipo de extensão um ficheiro está a utilizar e oq faz
- Possível listar compilar e executar comandos
- Diferencia ficheiros com o mesmo nome mas extensões diferentes
- Ficheiros apenas executáveis podem ser executados
- Ficheiros não compiláveis e não executáveis são identificados e dito o tipo de extensão que possuem

pontos fracos-

- Não é possível compilar mais do que um ficheiro ao mesmo tempo
- Não é possível compilar ficheiros que estejam na lista de ficheiros compiláveis
- Não é possível detectar mais de um argumento
- Não é possível compilar e executar um ficheiro ao mesmo tempo
- devido a utilizar a função chat as vezes ocorre segmentation file quando utilizando a opção -ai

Chatroom

No chatroom é possível criar um quarto ou juntar-se a outros utilizadores, numa comunicação peer-to-peer utilizando sockets. Na criação do quarto é possível definir o número máximo de pessoas que se pode conectar, e se é privado ou público. Se for privado, o utilizador terá que digitar uma password para entrar. Quando um utilizador entra, "autentica-se" com o servidor e tem acesso às últimas x mensagens enviadas nesse quarto, sendo que o número default para x é 10.

Pontos fortes

- Possível criar chats protegidos por passwords
- Possível definir número máximo de utilizadores

- Cada utilizador tem uma cor diferente
- Existe histórico de mensagens que é partilhado com um usuário quando entra.
- `epool()` é utilizado para esperar atividade nos fd das conexões
- Multithreading para gerir a entrada de pessoas no quarto.

Pontos fracos

- Não existe encriptação
- Não é possível a partilha de ficheiros.
- Implementação de mensagens não é bem feita
- Quando o dono do quarto se desconecta todos são desconectados.
- Todos os participantes precisam de estar na mesma rede, ou pelo menos ter regras de port Forwarding no router.
- Não é possível kickar/banir utilizadores
- Memória não é gerida da melhor maneira

Visualização de gráficos

Visualização de gráficos utiliza a biblioteca “ncurses” para desenhar um gráfico no terminal.

Pontos fortes

- Visualização clara dos gráficos
- Gráficos calculados em tempo real (sendo possível facilmente aceitar inputs de utilizadores)

Pontos fracos

- Não é possível inserir gráficos customizados

Integração com AI

O utilizador é capaz de fazer pedidos a um chatbot, graças à integração do Gemini AI.

Pontos fortes

- Pedidos feitos à API são rápidos

Pontos fracos

- Não é possível cancelar pedidos.
- Memory leak algures resultando em segmentation fault esporadicamente

Game of Life

Desenvolvido por John Conway, o “Game of Life” foi criado com intuito de simular a vida de “seres vivos” utilizando regras simples.

Pontos fortes

- Conway's Game of life

Pontos fracos

- Às vezes o programa não deteta corretamente o rato

Autocorreção

A autocorreção não conseguimos implementar, seria preciso alterar a arquitetura inteira do LEX, e já não tínhamos tempo de o fazer.

Bibliografia

Desenvolvendo um shell baseado em Linux – Acervo Lima. (n.d.). Acervo Lima. Retrieved March 3, 2024, from <https://acervolima.com/desenvolvendo-um-shell-baseado-em-linux/>