

UNIVERSIDADE EUROPEIA
IADE FACULDADE DE DESIGN, TECNOLOGIA E COMUNICAÇÃO
ENGENHARIA INFORMÁTICA

SHELLX

Desenvolvimento de um Shell para Linux

[HTTPS://GITHUB.COM/LESLYGT/SHELL-SO.GIT](https://github.com/leslygt/shell-so.git)

Djaline Cortez

20210355

Lesly Gombet

20210355

2. Descrição do Problema

O uso de um Shell no sistema operativo Linux é uma parte fundamental da interação do utilizador com o sistema, o que facilita a execução de comandos e programas, a manipulação de arquivos e processos, e o controle do sistema. No entanto, os shells padrão geralmente oferecem apenas funcionalidades básicas, deixando espaço para problemas então, o nosso objetivo com este projeto é melhorar a experiência do utilizador trazendo novas funcionalidades que resolvem estes problemas ou apenas transformam o Shell numa versão mais aprimorada e eficiente.

A motivação deste projeto surge da necessidade de criar um Shell mais avançado e adaptado às necessidades dos utilizadores, de forma a oferecer funcionalidades adicionais que simplifiquem tarefas comuns e melhorem a produtividade.

Casos de uso:

1º Caso:

1. O utilizador abre o ShellX.
2. Digita o comando específico para listar arquivos modificados recentemente em um diretório.
3. O ShellX exibe uma lista dos arquivos que foram modificados dentro de um intervalo de tempo determinado.

2º Caso:

1. O utilizador abre o ShellX.
2. Executa um comando no ShellX.
3. Após a conclusão do comando, o ShellX exibe uma notificação ou mensagem na tela do usuário informando que o comando foi concluído com sucesso.

3. Descrição da solução a implementar

1. Descrição genérica da solução:

Para enriquecer as funcionalidades básicas do shell, pretendemos implementar as seguintes novas funcionalidades:

- Comandos para listar todos os arquivos modificados recentemente num diretório.
- Armazenamento das três últimas janelas de comando executadas.
- Autocompletar comandos baseados nos comandos disponíveis e nos arquivos/diretórios presentes no sistema.
- Pesquisa, edição e execução de comandos anteriores.
- Exibição de notificações ou mensagens de conclusão após a execução de um comando.

O ShellX é um shell avançado desenvolvido para sistemas operativos Linux, projetado para oferecer uma experiência de utilizador aprimorada e funcionalidades adicionais em comparação com os shells padrão disponíveis. A sua arquitetura modular e sua implementação cuidadosa permitem uma ampla gama de funcionalidades, tornando-o numa ferramenta poderosa e versátil para interação com o sistema operativo.

O ShellX é composto por vários módulos funcionais, cada um responsável por uma parte específica da funcionalidade do shell, como análise sintática, execução de comandos, autocompletar, histórico de comandos e exibição de notificações. Esses módulos são projetados de forma independente, facilitando a extensibilidade e a manutenção do código.

Uma das características principais do ShellX é sua capacidade de armazenar todos os comandos executados em cada janela/terminal, proporcionando aos usuários acesso fácil e rápido a comandos

anteriores. Além disso, o ShellX oferece sugestões de autocompletar para comandos e nomes de arquivos/diretórios, tornando a entrada de comandos mais eficiente e intuitiva.

O ShellX também fornece notificações ou mensagens de conclusão após a execução de um comando, fornecendo feedback imediato ao utilizador sobre o resultado da operação. Isso ajuda a melhorar a interatividade do shell e a experiência geral do utilizador.

2. Enquadramento nas Áreas da Unidade Curricular:

Enquadra-se na UC por abordar conceitos fundamentais de interação do utilizador com o sistema operativos e práticas de programação em C.

3. Requisitos Técnicos:

- Conhecimento de linguagem C.
- Familiaridade com o sistema operativo Linux.
- Capacidade de trabalhar com estruturas de dados e manipulação de arquivos.

4. Arquitetura da Solução (temporária):

Interface: Responsável pela interação com o usuário, exibindo prompts e recebendo entradas de comandos.

Módulo de Análise Sintática: Analisa os comandos inseridos pelo usuário para identificar sua estrutura sintática e separar os comandos, argumentos e opções.

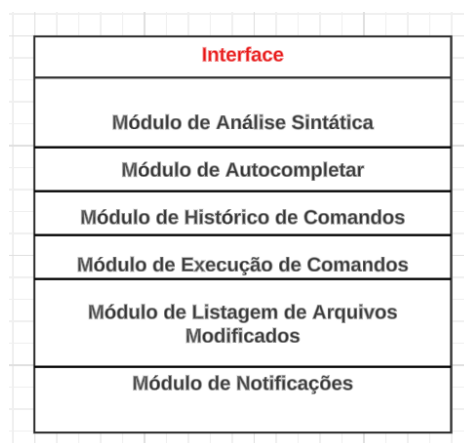
Módulo de Autocompletar: Oferece sugestões de autocompletar comandos baseados nos comandos disponíveis e nos arquivos/diretórios presentes no sistema.

Módulo de Histórico de Comandos: Mantém o histórico de comandos executados pelo usuário, permitindo pesquisa, edição e execução de comandos anteriores.

Módulo de Execução de Comandos: Executa os comandos fornecidos pelo usuário, chamando os binários associados e gerenciando sua execução.

Módulo de Listagem de Arquivos Modificados: Lista os arquivos modificados recentemente em um diretório específico.

Módulo de Notificações: Exibe notificações ou mensagens de conclusão após a execução de um comando



5. Tecnologias a Utilizar:

- IDE: VS Code
- Sistema Operacional: Linux
- VM: Ubuntu ou Vmware
- Ferramentas de Desenvolvimento: GCC, Makefile

- Controle de Versão: Git, Github

4. Planeamento e Calendarização

SHELLX

Lesly Gombet | Djaline Cortez

	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8
Pesquisa sobre interpretação de comandos em Linux								
Estudar as ferramentas e bibliotecas existentes que podem ser úteis								
Definir a estrutura básica do interpretador de comandos e as funcionalidades necessárias para cada uma das especificações fornecidas								
Projetar a interface do usuário e as interações com o sistema de arquivos.								
Começar implementando o interpretador de comandos básico, permitindo a execução de comandos simples do sistema operacional.								
Implementar os comandos para listar arquivos modificados recentemente em um diretório.								
Criar a funcionalidade de armazenamento das três últimas janelas de comando executadas.								
Desenvolver a funcionalidade de exibição de notificações ou mensagens de conclusão após a execução de um comando.								
Adicionar a funcionalidade de autocompletar comandos baseados nos comandos disponíveis e nos arquivos/diretórios presentes no sistema.								
Implementar a pesquisa, edição e execução de comandos anteriores.								
Realizar testes abrangentes para garantir que todas as funcionalidades estejam funcionando conforme o esperado.								

Legenda:

 Lesly

Djaline

6. Bibliografia

<https://www.gnu.org/software/bash/manual/bash.html>
<https://man7.org/linux/man-pages/man2/stat.2.html>
<https://chat.openai.com/>
https://youtu.be/zskFbzsZu8s?si=m_ppNogHGGVCrvs8