



RIOINFSIOP2 -Sistemas operacionais(2023 .2 - 912-A)

INSTITUTO FEDERAL DE ALAGOAS - RIO LARGO

REBECA CAETANO DOS SANTOS

JACIANE BARBOSA DA SILVA SANTOS

THAYSLANE CECÍLIA CONCEIÇÃO NASCIMENTO

AMAZON

Rio Largo, Alagoas

2023



RIOINFSIOP2 -Sistemas operacionais(2023 .2 - 912-A)

INSTITUTO FEDERAL DE ALAGOAS - RIO LARGO

REBECA CAETANO DOS SANTOS

JACIANE BARBOSA DA SILVA SANTOS

THAYSLANE CECÍLIA CONCEIÇÃO NASCIMENTO

AMAZON

Atividade apresentada curso Integrado ao ensino médio em Informática do Instituto Federal de Alagoas - Campus Rio Largo, como requisito relatório sobre código em shell script sobre a amazon da disciplina de Sistemas operacionais.

Docente: Pablo Gustavo Soares Tiburcio

Rio Largo, AL

2023

Sumário

Introdução.....	4
Amazon Logistics.....	5
Problemas.....	5
Shell Script.....	5

Introdução

A Amazon é uma gigante empresa de comércio eletrônico e tecnologia sediada nos Estados Unidos. Ela oferece uma variedade extensa de produtos, incluindo livros, eletrônicos, roupas, eletrodomésticos, brinquedos, entre outros. Além do comércio varejista, a Amazon também está envolvida em serviços de streaming, computação em nuvem (Amazon Web Services), inteligência artificial e dispositivos eletrônicos, como os populares leitores de e-books Kindle e os assistentes de voz da linha Echo.

Amazon Logistics

Pelo FBA - Logística da Amazon, você envia seus produtos para o depósito designado do marketplace e lá eles são classificados e armazenados. Quando um cliente efetua uma compra, a Amazon envia o item e faz todo o atendimento no pós-venda, inclusive se houver a necessidade de troca.

Problemas

A maioria das reclamações são sobre produtos não recebidos, cobranças indevidas, atraso na entrega. Como resolver?

Através de um script complexo em shell veremos a interação e a resolução de uma forma mais **rápida**(como todos nós queremos quando um pedido não atende nossas expectativas) e simples utilizando algumas estruturas: repetição, funções e condições.

Shell Script

```
main:~$ cat main.bash
1 #!/bin/bash
2
```

Indica que o script será executado usando o interpretador Bash.

```
3 # Função para acompanhar a entrega
4 acompanhar_entrega() {
5     dialog --inputbox "Digite o ID do produto para acompanhar a entrega:" 8 40 2>tempfile
6     id_produto=$(cat tempfile)
7     rm -f tempfile
8 }
```

`acompanhar_entrega()`: Define uma função chamada `acompanhar_entrega`.

`dialog`: Exibe um diálogo para o usuário inserir o ID do produto.

`tempfile`: Armazena o ID do produto em um arquivo temporário.

`id_produto=$(cat tempfile)`: Lê o conteúdo do arquivo temporário e atribui à variável `id_produto`.

rm -f tempfile: Remove o arquivo temporário.

```
8
9  # Lógica para confirmar a entrega ou informar sobre a não entrega
10 dialog --yesno "O produto com ID $id_produto foi entregue?" 8 40
11 resposta=$?
12
13 if [ $resposta -eq 0 ]; then
14     dialog --msgbox "Entrega confirmada para o produto com ID $id_produto." 8 40
15 else
16     dialog --yesno "Deseja solicitar um reembolso para o produto com ID $id_produto?" 8 40
17     resposta_reembolso=$?
18
19     if [ $resposta_reembolso -eq 0 ]; then
20         dialog --msgbox "Reembolso solicitado para o produto com ID $id_produto." 8 40
21         # Lógica para solicitar reembolso
22     else
23         dialog --msgbox "Acompanhe novamente mais tarde ou entre em contato para mais informações." 8 40
24     fi
25 fi
26 }
27
```

dialog de Sim/Não e Variável de Resposta: Pergunta ao usuário se o produto foi entregue e armazena a resposta em resposta.

Estrutura de Controle if: Verifica se a resposta é 0 (Sim). Se verdadeiro, exibe uma mensagem de confirmação.

else: Se a resposta não for 0, apresenta outra pergunta ao usuário.

Aninhamento de if: Um segundo bloco if aninhado dentro do bloco else.

Lógica para Reembolso: Se a resposta ao segundo dialog for 0 (Sim), exibe uma mensagem de reembolso. Pode incluir lógica adicional para solicitar o reembolso.

Mensagem Alternativa: Se a resposta ao segundo dialog não for 0, exibe uma mensagem alternativa.

```

28 # Função para solicitar aumento de prazo ou prioridade de entrega
29 ajustar_entrega() {
30     dialog --inputbox "Digite o ID do produto para ajustar a entrega:" 8 40 2>tempfile
31     id_produto=$(cat tempfile)
32     rm -f tempfile
33 }

```

Definição de Função (ajustar_entrega): Define uma função chamada ajustar_entrega.

dialog para entrada de dados: Exibe um diálogo para o usuário inserir o ID do produto.

Leitura de Arquivo Temporário (tempfile): Armazena o ID do produto em um arquivo temporário.

Atribuição de Variável (id_produto=\$(cat tempfile)): Lê o conteúdo do arquivo temporário e atribui à variável id_produto.

Remoção de Arquivo Temporário (rm -f tempfile): Remove o arquivo temporário.

```

34 # Lógica para oferecer opções de ajuste
35 dialog --menu "Escolha uma opção:" 10 40 3 1 "Aumentar prazo de entrega" 2 "Solicitar prioridade de entrega" 3 "Cancelar" 2>tempfile
36
37 opcao=$(cat tempfile)
38 rm -f tempfile
39

```

dialog de Menu: Apresenta um menu ao usuário com três opções e armazena a escolha em um arquivo temporário.

Leitura de Arquivo Temporário (opcao=\$(cat tempfile)): Lê a escolha do usuário do arquivo temporário e a atribui à variável opcao.

Remoção de Arquivo Temporário (rm -f tempfile): Remove o arquivo temporário.

```

39
40 case $opcao in
41     1)
42         dialog --msgbox "Prazo de entrega aumentado para o produto com ID $id_produto." 8 40
43         # Lógica para aumentar prazo de entrega
44         ;;
45     2)
46         dialog --msgbox "Prioridade de entrega solicitada para o produto com ID $id_produto." 8 40
47         # Lógica para solicitar prioridade de entrega
48         ;;
49     3)
50         dialog --msgbox "Operação cancelada." 8 40
51         ;;
52     *)
53         dialog --msgbox "Opção inválida." 8 40
54         ;;
55 esac
56 }

```

case: Avalia o valor da variável opcao e executa o bloco correspondente.

Opções do case: Cada opção (1, 2, 3, *) tem um bloco associado. O bloco *) funciona como um "default" para opções inválidas.

```
57
58 # Menu principal
59 while true; do
60     dialog --menu "Escolha uma opção:" 12 40 5 1 "Acompanhar entrega" 2 "Solicitar ajuste de entrega" 3 "Sair" 2>tempfile
61
62     escolha=$(cat tempfile)
63     rm -f tempfile
64
```

Loop while true: Cria um loop infinito para exibir repetidamente o menu principal.

dialog de Menu Principal: Apresenta um menu ao usuário com três opções e armazena a escolha em um arquivo temporário.

Leitura de Arquivo Temporário (escolha=\$(cat tempfile)): Lê a escolha do usuário do arquivo temporário e a atribui à variável escolha.

Remoção de Arquivo Temporário (rm -f tempfile): Remove o arquivo temporário.

```
64
65     case $escolha in
66         1)
67             acompanhar_entrega
68             ;;
69         2)
70             ajustar_entrega
71             ;;
72         3)
73             dialog --msgbox "Saindo do script. Até mais!" 8 40
74             exit 0
75             ;;
76         *)
77             dialog --msgbox "Opção inválida." 8 40
78             ;;
79     esac
80 done
81
```

Essa parte do código é responsável por determinar qual ação tomar com base na escolha do usuário (`\$escolha`). Cada número no menu tem uma ação associada, e o bloco * atua como um caso padrão para lidar com escolhas inválidas.