

UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA

VIVIAN MARTINS MOURA

RELATÓRIO PBL

FEIRA DE SANTANA

2025

1. INTRODUÇÃO

1.1 Resumo do problema

A cidade de TechVille quer melhorar seu sistema de transporte público para torná-lo mais eficiente e acessível. Um desafio é gerenciar automaticamente os saldos dos cartões de transporte, permitindo que os usuários embarquem sem problemas e permitindo que o sistema acompanhe recargas e uso. Um sistema piloto deverá ser desenvolvido para registrar recargas, descontar automaticamente o valor das passagens e verificar o saldo. Existem três tipos de usuários: padrão, estudante/idoso e social, cada um com regras de desconto no valor da passagem. A administração do transporte precisa de relatórios para analisar o uso do serviço e planejar melhorias. A motivação principal é modernizar o transporte, proporcionando praticidade e segurança aos usuários, além de ajudar a gestão municipal a otimizar investimentos. O sistema deve ser flexível, podendo ser adaptado para outras cidades.

1.2 Descrição breve da solução

Um código bem estruturado facilitará manutenções futuras e expansões. Por isso, eu desenvolvi esse sistema utilizando um código em python na versão mais atual, a versão 3.13.2, no Visual Studio Code e implementei tudo que foi solicitado.

1. METODOLOGIA

2.1 Processo de construção de conhecimento nas sessões tutoriais

Durante as sessões tutoriais foram abordados diversos assuntos, e uma das primeiras decisões que obtivemos em conjunto foi o looping que teríamos que implementar no menu, já que o programa deveria continuar funcionando até o usuário desejar sair. Nessas sessões também obtivemos maneiras diferentes de resolução do problema, por exemplo, tivemos pessoas que criaram um sistema com 130 linhas de código e outras que criaram com 300 linhas.

2.2 Definição de requisitos

O sistema precisa permitir a definição do valor da passagem logo que ele for inicializado, para poder ser utilizado em apresentações em outras cidades. Ele também precisa registrar recargas, conferir os descontos das passagens dependendo do tipo de usuário (idoso/estudante, padrão, social), descontar do saldo quando alguma compra de passagem é feita, além de verificar se o saldo é suficiente para a compra da passagem, informando assim ao usuário se ele possui saldo suficiente para embarcar ou se ele precisa fazer uma recarga para conseguir comprar a passagem. Além disso, o sistema deve permitir que o administrador peça um relatório a qualquer momento, exibindo na tela as seguintes informações:

- Quantidade de recarga total e o número de recargas realizadas por cada usuário;
- Quantidade de passagens usadas por cada tipo de usuário;
- Valor total gasto com passagens por cada tipo de usuário;
- Saldo restante a ser utilizado por cada tipo de usuário.

2.3 Descrição de alto nível

Meu programa é iniciado solicitando primeiramente o valor da passagem, com um input, e após isso ocorre a criação das variáveis que serão utilizadas ao longo do programa. Após isso, eu criei o menu para a escolha do tipo de usuário, que são: idoso/estudante (paga metade do valor da passagem), padrão (paga o valor cheio da passagem), e social (paga 20% do valor da passagem). Após o usuário escolher sua categoria, o preço da passagem já é alterada para o preço com o desconto, como mostra a figura abaixo:

```
categorias=print(f'Categorias: {idoso}=idoso ou estudante / {padrao}=padrão / {social}=social')
categorias=5
usuario=int(input('Digite qual a sua categoria:'))
if usuario==idoso:
    passagem= passagem/2
    print(f'Sua passagem com desconto custa {passagem} reais')
elif usuario==padrao:
    passagem=passagem
    print(f'Sua passagem custa {passagem} reais')
elif usuario==social:
    passagem=passagem/5
    print(f'Sua passagem com desconto custa {passagem} reais')
```

Após isso, eu criei um loop "while" para enquanto o usuário não digitar o número 3 (que corresponde a 'sair'), ele continue rodando. E criei um menu de categoria antes disso, onde apresenta 5 opções para o usuário decidir o que ele deseja fazer, como mostra a figura abaixo:

```
recarga=1
cpassagem=2
sair=3
relat=4
categorias=5

print(f'{recarga}=recarga / {cpassagem}=comprar passagem / {sair}=sair do sistema / {relat}=relatório / {categorias}=mudar de usuário')
us=int(input('Digite o que você deseja fazer em seguida:'))
```

Dentro do loop eu criei 5 "if's, 1 para cada opção disponível, para recarregar, para comprar passagem, para sair do sistema, para emitir o relatório e para mudar o tipo de usuário. Ou seja, se o usuário digitar 1 ele deseja recarregar, então irá aparecer um input para ele digitar quanto dinheiro ele deseja colocar no seu saldo inicial (0 reais). Dentro do "if" principal tem outros "if's para cada tipo de usuário, por exemplo, se o usuário for do tipo padrão e recarregar 20 reais esse valor vai ser adicionado ao

saldo da variável criada para o saldo da categoria padrão especificamente, como mostra na imagem abaixo:

```
if us==1: #Criação do if para o caso do usuário querer recarregar, que é represe
    r=int(input('Digite o valor que você deseja adicionar no seu saldo:'))
    saldo=(saldo+r)
    print(f'Seu saldo atual após a recarga é de {saldo} reais')
    recargatotal+=1
    valorrecargatotal+=r
    if usuario==1 and us==1:
        recargatotal1+=1
        valorrecargaidoso+=r
        saldoidoso=saldo
    elif usuario==2 and us==1:
        valorrecargapadrao+=r
        recargatotal2+=1
        saldopadrao=saldo
    elif usuario==3 and us==1:
        valorrecargasocial+=r
        recargatotal3+=1
        saldosocial=saldo
```

Ou seja, cada categoria tem uma variável diferente para armazenar o valor da recarga daquele tipo de usuário e para armazenar a quantidade de vezes que aquele tipo de usuário recarregou. Tudo isso para poder exibir essas informações no relatório, caso este seja solicitado. Caso o usuário digite 2, ele deseja comprar passagem, aí dentro desse "if" eu adicionei outros "if"s para verificar se o saldo atual do usuário é suficiente para realizar a compra ou não, se for suficiente ele irá realizar a compra e armazenará a quantidade de vezes que cada tipo de usuário comprou, se o saldo não for suficiente ele irá printar 'Seu saldo atual é insuficiente!'. Como mostra a imagem abaixo, usando o usuário 1 (idoso/estudante) como exemplo:

```
if usuario==1 and saldo>=passagem:
    saldo=(saldo-passagem)
    print(f'Sua compra foi feita com sucesso! Seu saldo restante é:{saldo} reais')
    quantdepassagem+=1
    passagemidoso=passagem
elif usuario==1 and saldo<passagem:
    print('Seu saldo atual é insuficiente!')
```

Caso o usuário digite 3 ele deseja sair então o programa fecha e para de rodar o loop. Caso o usuário digite 4 ele deseja ver o relatório, que contêm todas as informações que foram armazenadas nas variáveis, e nesse relatório é onde mostra a quantidade total de recargas feitas, valor total gasto com passagem, entre outras coisas que serão mostradas na imagem abaixo:

```

print('Relatório:')
print(f'Valor total das recargas: {valorrecargatotal} reais')
print(f'Valor total das recargas na categoria idoso/estudante:{valorrecargaidoso} reais')
print(f'Valor total das recargas na categoria padrão: {valorrecargapadrao} reais')
print(f'Valor total das recargas na categoria social: {valorrecargasocial} reais')
print(' ')
print(f'Quantidade de recargas feitas:{recargatotal}')
print(f'Quantidade de recargas feitas na categoria idoso/estudante: {recargatotal1}')
print(f'Quantidade de recargas feitas na categoria padrão: {recargatotal2}')
print(f'Quantidade de recargas feitas na categoria social: {recargatotal3}')
print(' ')
print(f'Quantidade de passagens usadas na categoria idoso/estudante: {quantdepassagem}')
print(f'Quantidade de passagens usadas na categoria padrão: {quantdepassagem1}')
print(f'Quantidade de passagens usadas na categoria social: {quantdepassagem2}')
print(' ')
if usuario==1 or usuario==2 or usuario==3:
    valorgastoidoso=(quantdepassagem*passagemidoso)
    print(f'Valor gasto com passagem na categoria idoso/estudante:{valorgastoidoso}')
    valorgastopadrao=(quantdepassagem1*passagempadrao)
    print(f'Valor gasto com passagem na categoria padrão:{valorgastopadrao}')
    valorgastosocial=(quantdepassagem2*passagemsocial)
    print(f'Valor gasto com passagem na categoria social:{valorgastosocial}')
print(' ')
if usuario==1 or usuario==2 or usuario==3:
    usuario1=print(f'Saldo restante na categoria idoso/estudante:{saldoidoso}')
    usuario2=print(f'Saldo restante na categoria padrão:{saldopadrao}')
    usuario3=print(f'Saldo restante na categoria social:{saldosocial}')

```

E caso o usuário digite 5 ele deseja trocar de categoria, então caso ele mude de categoria o valor da passagem inicial terá um desconto novo que será alterado assim que ele escolher qual categoria ele quer, como mostra na imagem abaixo:

```

elif us==5: #criação do if para o caso do usuário querer mudar de categoria
    print(f'Categorias: {idoso}=idoso ou estudante / {padrao}=padrão / {social}=social')
    usuario=int(input('Digite qual usuário você se encaixa:'))
    passagem=passageminicial
    saldo=0
    print('Saldo inicial=0 reais')
    if usuario==1:
        passagem=passagem/2
    if usuario==2:
        passagem=passagem
    if usuario==3:
        passagem=passagem/5

```

2.4 Ordem de codificação

Eu comecei pensando em criar inicialmente o menu inicial e o input para adicionar o valor da passagem, então comecei fazendo isso, criei os três tipos de usuários e coloquei o input para a pessoa digitar qual categoria ela pertencia. Logo após isso pensei como faria para o sistema funcionar até o usuário desejar parar, e percebi que teria que usar o “while” para isso, então comecei criando um loop que continuaria rodando até o usuário desejar sair, e dentro desse loop fui criando vários “if’s. Fiz essa parte principal em uma tarde. Porém, após uma semana eu tive uma sessão tutorial e percebi que meu código apresentava alguns erros então assim que pude fiz alterações para consertá-los. Por exemplo, meu código inicial dava a opção do usuário digitar o seu saldo inicial, mas após a sessão eu percebi que o saldo inicial

de todas as categorias sempre será zero, então alterei isso. E após esses ajustes meu código ficou pronto, então uma semana depois fiz a última parte do trabalho que era comentar o código e adicionar o “não plágio” no início do código.

2. RESULTADOS E DISCUSSÕES

3.1 Manual de uso

3.1.1 Como utilizar esse programa?

- O programa solicita ao usuário o valor da passagem e sua categoria (idoso/estudante, padrão ou social). Em seguida, apresenta um menu onde o usuário pode recarregar o saldo, comprar passagens, visualizar um relatório ou mudar de categoria. O loop continua até que o usuário escolha sair do sistema.

3.1.2 Qual é o conjunto de dados de entrada válido para o correto funcionamento do programa?

- Valor da passagem: Deve ser um número real positivo.
- Categoria do usuário: Deve ser um número inteiro correspondente a uma das opções (1 = idoso/estudante, 2 = padrão, 3 = social).
- Ações do menu: O usuário deve escolher entre as opções numéricas disponíveis (1 = recarga, 2 = comprar passagem, 3 = sair, 4 = relatório, 5 = mudar de usuário).
- Valores de recarga: Devem ser números inteiros ou decimais positivos.

3.1.3 Quais são as saídas do programa?

- Mensagens exibindo o saldo atualizado após recargas.
- Confirmações de compra de passagens ou avisos de saldo insuficiente.
- Um relatório detalhado contendo o total de recargas, quantidade de passagens usadas, valor gasto por categoria e saldo restante.

3.1.4 Como são geradas as saídas (processamento para obtê-las)?

- O programa verifica a categoria do usuário para aplicar o desconto correto no valor da passagem.
- Quando o usuário recarrega, o saldo é atualizado e armazenado conforme a categoria.
- Ao comprar uma passagem, o saldo é verificado e, se suficiente, é descontado conforme o valor da passagem.

- O relatório é gerado somando os valores de recarga e passagens compradas, exibindo os totais organizados por categoria.

3.1.5 Quais os testes efetuados (durante e ao final do desenvolvimento) e quais os resultados obtidos?

- Testei todas as possibilidades possíveis, com todos os tipos de usuário. Tentei comprar passagem, recarregar, ver relatório, mudar categoria, sair, entre outros. Tudo está funcionando perfeitamente.

3.1.6 Quais erros ocorreram nos testes? Em que situação o seu programa não funcionaria?

- Apenas não consegui implementar a verificação das entradas caso o usuário digite uma letra ou um número negativo, então caso isso ocorra o meu código não irá funcionar.

3. CONCLUSÃO

- Resumindo, acredito que o código no geral está funcionando muito bem, apenas a verificação da entrada que não foi possível implementar, porém os principais objetivos foram cumpridos.

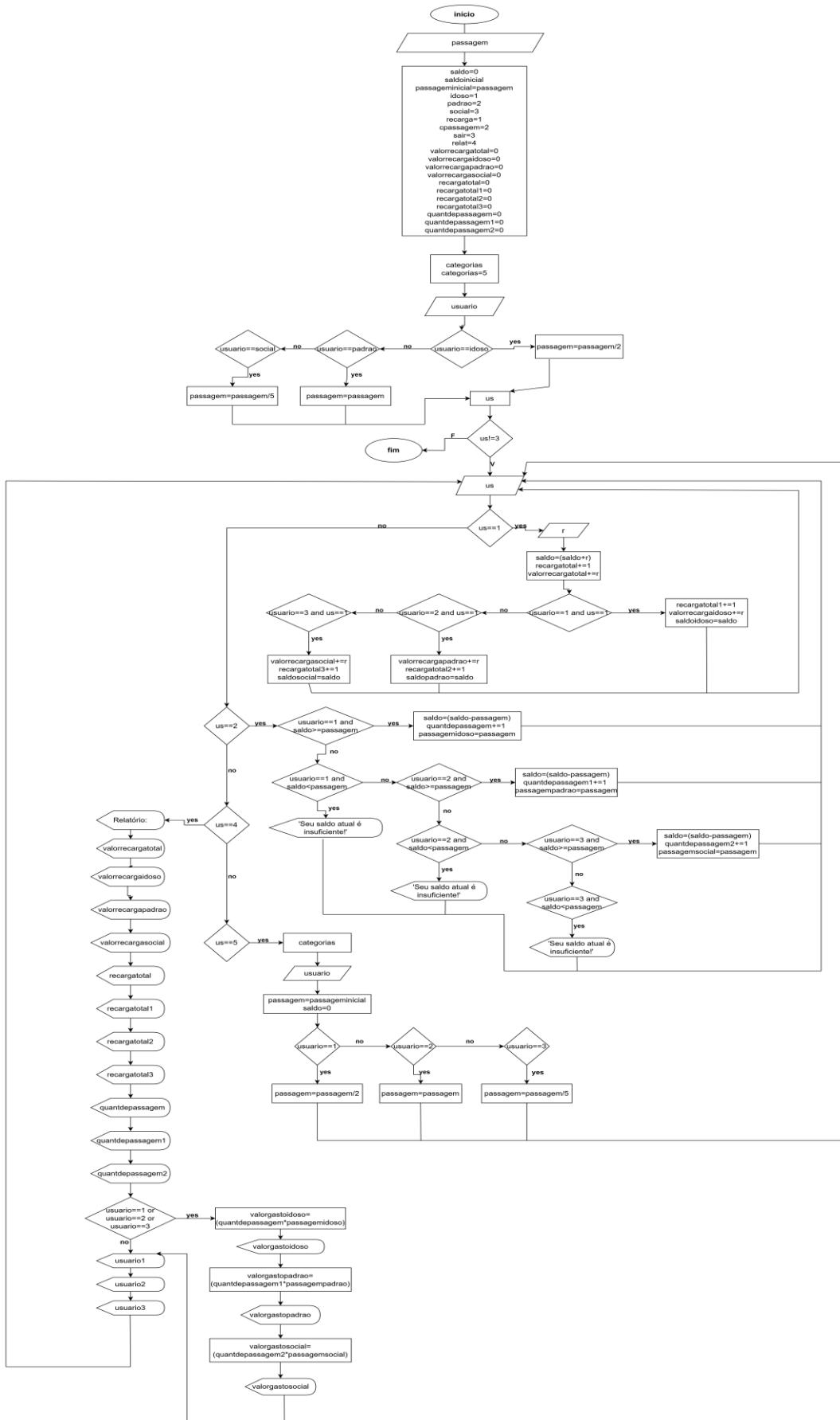
4. BIBLIOGRAFIA CONSULTADA

CURSO EM VÍDEO. *Python - Aula 01 - Introdução e instalação*. YouTube, 25 out. 2019. Disponível em: <https://youtu.be/bG3YuzilcZI>. Acesso em: 12 mar. 2025.

CIÊNCIA DE DADOS COM PYTHON. *Laço while com contador no Python*. YouTube, 2 nov. 2022. Disponível em: https://youtu.be/kfct_HXiFxg. Acesso em: 12 mar. 2025.

REFATORANDO. *Curso de Python para iniciantes #09 - Loops: While e For*. YouTube, 8 dez. 2020. Disponível em: https://youtu.be/o-1ciQ_l8-4. Acesso em: 10 mar. 2025.

5. FLUXOGRAMA



Link:

<https://1drv.ms/u/c/a09ee6ae0d25432e/ExaTlqOvY89IgxDUGqxg78ABv3mz7EQCTatHn9kQNDLpFw?e=pu7QO7>