



UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Disciplinas SSC0501 e SSC0502

Curso de Bacharelado de Sistema da Informação

DOCUMENTAÇÃO TRABALHO PRÁTICO

Arthur Gagliardi Azorli

Nº USP 16855452



São Carlos

Mai./2025



Sumário

- **Descrição do Software**
 - *Formato do arquivo externo*
 - **Menu Principal**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*
 - **Cadastro de Usuário**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*
 - **Cadastro de Múltiplos Usuários**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*
 - **Seleção de Usuário**
 - *Descrição*
 - *Pseudocódigo*
 - **Deletar Usuário**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*
 - **Buscar Usuário**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*
 - **Transferência entre Usuários**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*
-



- **Saque de Valor**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*

 - **Deposito de Valor**
 - *Descrição*
 - *Fluxograma*
 - *Pseudocódigo*

 - **Salvar/Carregar Dados do Arquivo**
 - *Descrição*
 - *Fluxograma Salvar*
 - *Fluxograma Carregar*

 - **Observações**
-



Descrição do Software

O Software em questão se trata de um sistema bancário, pertencente a rede bancária UAI BANK, ao qual se responsabilizará pela gestão de usuários do banco. O Programa será desenvolvido na Linguagem C, fornecendo uma interface em linha de comando para interações do usuário, como cadastro de um ou mais usuários, pesquisa de usuário em específico, remoção de usuário, transferência bancária entre dois usuários, saque e depósito de valores. Além de utilizar-se de alocação dinâmica de memória para armazenar as informações dos usuários e passá-los para um arquivo externo, sendo elas, o id único gerado automaticamente, nome completo, idade e saldo atual.

Deve, ainda, incluir os seguintes requisitos exigidos pelo UAI BANK:

- O campo nome deverá ter no máximo 100 caracteres;
- A saída padrão deverá conter informações sobre o resultado de cada uma das transações solicitadas.
- O arquivo final deverá ter o mesmo formato definido neste documento, bem como a listagem de usuários esperada na ordem correta.
- O único campo que deverá ter valores únicos é o id.
- Nenhum usuário poderá ter saldo negativo.

O Programa de Desenvolvimento será o CLion da JetBrains, no plano gratuito para sem fins de corporativos.

Formato do Arquivo Externo

O Arquivo externo será um documento de texto, extensão .txt, onde será armazenado as informações da seguinte maneira:

<ID><COLUMN_DELIM><NOME><COLUMN_DELIM><IDADE><COLUMN_DELIM><SALDO>

Sendo, <ID>, o identificador único gerado automaticamente; <NOME>, o nome completo do usuário; <IDADE>, a idade do usuário, apenas o número; <SALDO>, o saldo atual do usuário, sem indicador de unidade de valor (R\$), contendo apenas números e casas decimais separada por vírgula (,), seguindo o padrão de valor decimal do Windows 11 em Português – BR; <COLUMN_DELIM>, o delimitador de cada coluna, ao qual será definido como “ ||&|| ”, para evitar conflitos internos.

O nome do arquivo será “file_save.txt”. E o arquivo responsável por armazenar o último id gerado, e apenas isso, será “id.txt”



Menu Principal

Descrição

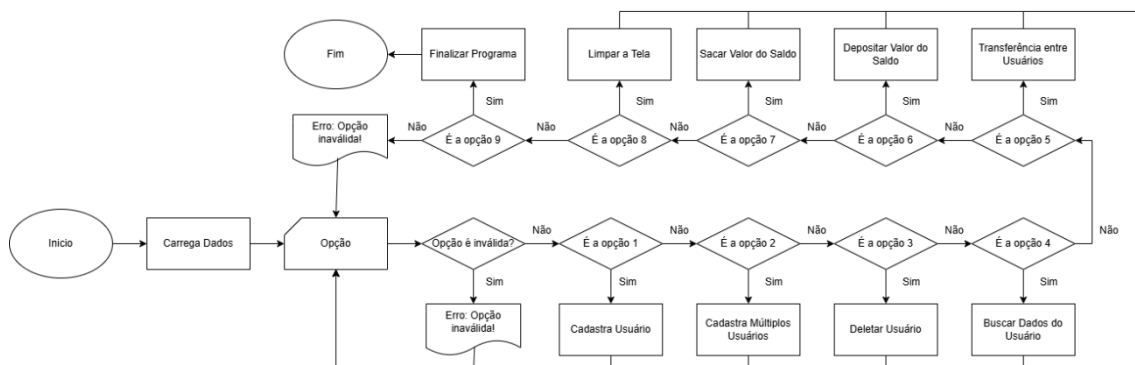
O menu principal será o meio de escolha de funcionalidade que o usuário executará, sendo assim, deve apresentar as opções que ele pode escolher sendo elas:

1. Cadastrar novo Usuário
2. Cadastrar múltiplos usuários
3. Deletar Usuário
4. Buscar por Usuário
5. Realizar Transferência
6. Realizar Deposito
7. Realizar Saque
8. Limpar tela
9. Finalizar Programa

O usuário escolherá a opção digitando o número da opção, como definido neste documento, e após terminar a funcionalidade deve apresentar novamente o menu de opções para a próxima funcionalidade, até finalmente o usuário digitar a opção de sair (9), ao qual finalizará o programa em execução.

Caso o usuário digite uma opção inexistente ou inválida deve exibir uma mensagem erro.

Fluxograma





Pseudocódigo

```
Menu_Principal

INICIO
    DECLARAR opcao
    CARREGAR_BANCO()
    ENQUANTO opcao != '9' FACA
        PRINT_MENU()
        LER opcao
        ESCOLHA opcao FACA
            CASO '1'
                CADASTRAR_USUARIO()
            FIMCASO
            CASO '2'
                CADASTRAR_MULTIPLOS_USUARIOS()
            FIMCASO
            CASO '3'
                DELETAR_USUARIO()
            FIMCASO
            CASO '4'
                BUSCAR_USUARIO()
            FIMCASO
            CASO '5'
                TRANSFERENCIA()
            FIMCASO
            CASO '6'
                DEPOSITO()
            FIMCASO
            CASO '7'
                SAQUE()
            FIMCASO
            CASO '8'
                LIMPAR_TELA()
            FIMCASO
            CASO '9'
                ESCREVER "Finalizando Sistema..."
                SALVAR_BANCO()
            FIMCASO
            CASO CONTRARIO
                ERROR_MESSAGE("Entrada invalida!")
            FIMCASO
        FIMESCOLHA
    PRESS_ENTER()
    FIMENQUANTO
FIM
```



Cadastrar Usuário

Descrição

O Cadastro do usuário irá ler:

- o nome completo do usuário com no mínimo 3 e no máximo 100 caracteres;
- a idade do usuário entre 16 e 120 anos;
- caso deseje, também lerá o saldo inicial do usuário, superior ou igual a zero, senão atribuirá o valor por default a zero.

Caso o usuário digite qualquer um desses valores de forma inválida, ele deve digitar novamente até digital corretamente, com exceção caso o usuário digite uma idade inferior a 16, onde deve apenas exibir uma mensagem de erro. Com isso também gerará automaticamente um identificado único (ID) para o usuário, e finalmente salvará os dados do usuário no arquivo externo.

Fluxograma





Pseudocódigo

```
Cadastrar_Usuario

INICIO
    DECLARAR nome, idade, saldo, opção, usaSaldo

    ENQUANTO VERDADEIRO FACA
        ESCRIVER "Digite o nome completo do usuario: "
        LER nome

        SE tamanho(nome) < 3 ENTAO
            ESCRIVER "Quantidade insuficiente de caracteres! Mínimo 3"
            CONTINUAR
        FIMSE

        SE tamanho(nome) > 100 ENTAO
            ESCRIVER "Excedeu o limite de caracteres!"
            CONTINUAR
        FIMSE

    PARAR
FIMENQUANTO

    ENQUANTO VERDADEIRO FACA
        ESCRIVER "Digite a idade do usuario: "
        LER idade

        SE idade > 120 OU idade < 0 ENTAO
            ESCRIVER "Idade inválida!"
            CONTINUAR
        FIMSE

        SE idade < 16 ENTAO
            ESCRIVER "Somente pessoas acima de 16 anos podem criar uma conta!"
            CONTINUAR
        FIMSE
    PARAR
FIMENQUANTO

    ENQUANTO VERDADEIRO FACA
        ESCRIVER "Deseja começar com um saldo inicial? (S/N): "
        LER opcao

        ESCOLHA opcao FACA
            CASO 'S'
            CASO 's'
                usaSaldo = VERDADEIRO
            FIMCASO
            CASO 'N'
            CASO 'n'
                usaSaldo = FALSO
            FIMCASO
            CASO CONTRARIO
                ESCRIVER "Opção inválida!"
                CONTINUAR
            FIMCASO
        FIMESCOLHA
    PARAR
FIMENQUANTO

    SE usaSaldo == VERDADEIRO ENTAO
        ENQUANTO VERDADEIRO FACA
            ESCRIVER "Digite o valor do saldo: "
            LER saldo
            SE saldo <= 0 ENTAO
                ESCRIVER "Quantia inválida!"
                CONTINUAR
            FIMSE
        FIMENQUANTO
        SENAO
            saldo <- 0
        FIMSE
    INIT_USER(nome, idade, saldo)
    ESCRIVER "Usuário criado com sucesso!"
FIM
```

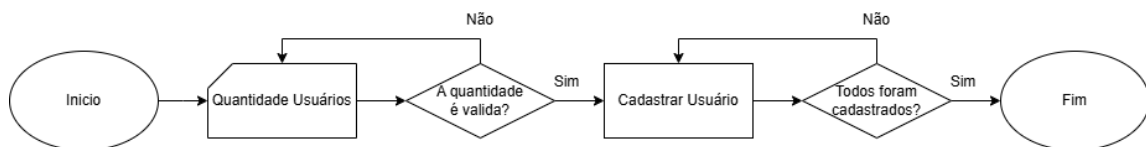



Cadastrar Múltiplos Usuário

Descrição

O Cadastro de Múltiplos Usuários deve ler uma quantidade acima de 0 e igual ou inferior 10, e fazer o cadastro dessa quantidade de usuários, seguindo o mesmo padrão da funcionalidade de cadastro de usuário.

Fluxograma



Pseudocódigo

```
Cadastrar_Multiplos_Usuarios

INICIO
    DECLARAR quantidade

    ENQUANTO VERDADEIRO FACA
        ESCRIVER "Digite a quantidade de usuários desejados (Máximo 10): "
        LER quantidade

        SE quantidade <= 0 ENTAO
            ESCRIVER "Quantidade inválida!"
            CONTINUAR
        FIMSE

        SE quantidade > 10 ENTAO
            ESCRIVER "Excedeu o limite de 10 usuários por vez!"
            CONTINUAR
        FIMSE

        PARAR
    FIMENQUANTO

    PARA i DE 0 A quantidade - 1 FACA
        CADASTRAR_USUARIO()
    FIMPARA
FIM
```



Selecionar Usuário

Descrição

A Seleção de usuário de usuário é uma função de utilidade que será usada pelas funcionalidades seguintes. Basicamente trata da leitura de um ID digitado pelo usuário, da verificação de valor válido e cujo usuário exista, assim retorna o ID verificado. Caso o valor seja inválido pede ao usuário que digite novamente o ID até inserir um valor válido. Caso não tenha usuários cadastrados termina a execução da função.

Obs.: O seu funcionamento em fluxograma é exemplificado nos fluxogramas que necessitam dessa funcionalidade.

Pseudocódigo

```
Selecionar_Usuario

INICIO
  SE A tamanho(listaUsuarios) == 0 ENTAO
    ESCRIVER "Nao ha usuarios cadastrados!"
    RETORNAR NULL
  FIM SE

  DECLARAR id
  ENQUANTO VERDADEIRO FAÇA
    ESCRIVER "Digite o ID do usuario desejado: "
    LER id
    PARA i DE 0 A tamanho(listaUsuarios) - 1 FAÇA
      DECLARAR usuario
      Usuario <- listaUsuarios[i]
      SE usuario != NULL E usuario.id == id ENTAO
        RETORNAR i
      FIM SE
    FIM PARA
  FIM ENQUANTO
  RETORNAR NULL
FIM
```

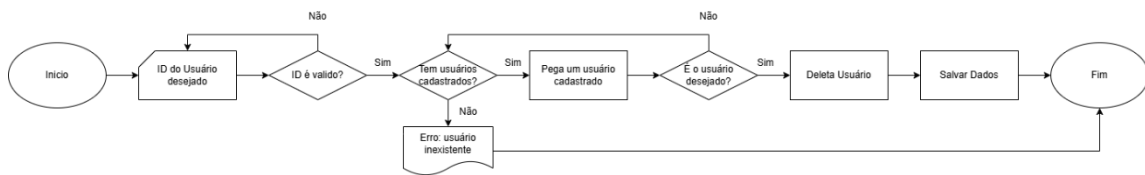


Deletar Usuário

Descrição

A Remoção do Usuário, deve selecionar um usuário por ID, usando a seleção de usuário, e retirar o usuário do arquivo externo. Caso não tenha usuários cadastrados ou ocorra um erro durante a remoção deve exibir uma mensagem de erro e terminar a execução.

Fluxograma



Pseudocódigo

```
Deletar_Usuario

INICIO

SE A tamanho(listaUsuarios) == 0 ENTAO
    ESCRIVER "Nao ha usuarios cadastrados!"
    RETORNAR NULL
FIM SE

DECLARAR indice
indice <- selecionarUsuario("Selecione um Usuario")

SE indice != NULL E removerValor(listaUsuarios, indice) ENTAO
    ESCRIVER "Usuario removido com sucesso!"
SENAO
    ESCRIVER "Usuario inexistente!"
FIM SE

SALVAR_DADOS_USUARIOS(listaUsuarios)

FIM
```

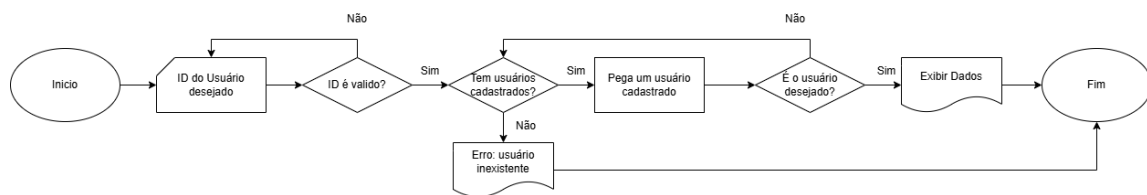


Buscar Usuário

Descrição

A Busca por Usuário, deve selecionar um usuário por ID, usando a seleção de usuário, e exibir o nome, idade e saldo do usuário. Caso não tenha usuários cadastrados deve exibir uma mensagem de erro e terminar a execução.

Fluxograma



Pseudocódigo

```
Buscar_Usuario

INICIO
    SE A tamanho(listaUsuarios) == 0 ENTAO
        ESCRIVER "Nao ha usuarios cadastrados!"
        RETORNAR NULL
    FIM SE

    DECLARAR indice
    indice <- selecionarUsuario("Selecione um usuario")

    SE indice != NULL ENTAO
        DECLARAR usuario;
        usuario <- listaUsuarios[indice]
        ESCRIVER "nome: ", usuario.nome
        ESCRIVER "idade: ", usuario.idade
        ESCRIVER "saldo: ", usuario.saldo
    SENAO
        ESCRIVER "Usuario Inexistente!"
    FIM SE
FIM
```

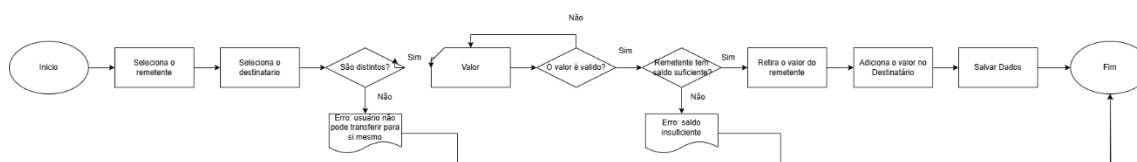


Transferência entre Usuários

Descrição

A Transferência entre Usuários, deve selecionar dois usuários por ID, usando a seleção de usuário, sendo eles respectivamente o Remetente e o Destinatário, além disso deve ler o valor, acima de zero, que será transferido do remetente para o destinatário e enfim transferi-lo. Caso não tenha usuários cadastrados, o remetente e o destinatário sejam o mesmo usuário ou o remetente não tenha o valor necessário, exibe uma mensagem de erro e termina a execução. Caso o valor seja inválido, deve solicitar ao usuário digitá-lo novamente até ele inserir um valor válido. Ao fim deve salvar os dados no arquivo externo.

Fluxograma





Pseudocódigo

```
Transfereir_Valor

INICIO
    SE A tamanho(listaUsuarios) == 0 ENTAO
        ESCREVER "Nao ha usuarios cadastrados!"
        RETORNAR NULL
    FIM SE

    SE tamanho(listaUsuarios) < 2 ENTAO
        ESCREVER "Nao ha usuarios o suficiente!"
        RETORNAR NULL
    FIM SE

    DECLARAR indiceRemetente
    indiceRemetente <- selecionarUsuario()

    DECLARAR indiceDestinatario
    indiceDestinatario <- selecionarUsuario()

    DECLARAR valor
    ENQUANTO VERDADEIRO FAÇA
        ESCREVER "Digite o valor da transferência:"
        LER valor

        SE valor <= 0 ENTAO
            ESCREVER "Quantia inválida!"
            CONTINUAR
        FIMSE
    FIMENQUANTO

    DECLARAR remetente
    remetente <- listaUsuarios[indiceRemetente]

    DECLARAR destinatario
    destinatario <- listaUsuarios[indiceDestinatario]

    SE remetente.saldo >= valor E remetente != destinatario ENTAO
        remetente.saldo <- remetente.saldo - valor
        destinatario.saldo <- destinatario.saldo + valor
        ESCREVER "Transferencia concluida com sucesso!"
    SENAO
        ESCREVER "Houve um erro durante a Transferencia!"
    FIM SE

    SALVAR_DADOS_USUARIOS(listaUsuarios)
FIM
```



Deposito de valor

Descrição

O deposito de valor, deve selecionar um usuário por ID, usando a seleção de usuário, ler um valor superior a zero, que deverá ser adicionado a seu saldo. Caso o valor seja inválido, deve solicitar ao usuário digitá-lo novamente até ele inserir um valor válido. Ao fim deve salvar os dados no arquivo externo.

Fluxograma





Pseudocódigo

```
Depositar_Valor

INICIO
    SE A tamanho(listaUsuarios) == 0 ENTAO
        ESCRIVER "Nao ha usuarios cadastrados!"
        RETORNAR NULL
    FIM SE

    DECLARAR índice
    indice <- selecionarUsuario()

    DECLARAR usuario
    usuario <- listaUsuarios[indice]

    DECLARAR valor
    ENQUANTO VERDADEIRO FAÇA
        ESCRIVER "Digite o valor do depósito:"
        LER valor
        SE valor <= 0 ENTAO
            ESCRIVER "Quantia inválida!"
            CONTINUAR
        FIMSE
    FIMENQUANTO

    usuario.saldo <- usuario.saldo + valor

    ESCRIVER "Deposito realizado com sucesso!"
    SALVAR_DADOS_USUARIOS(listaUsuarios)
FIM
```

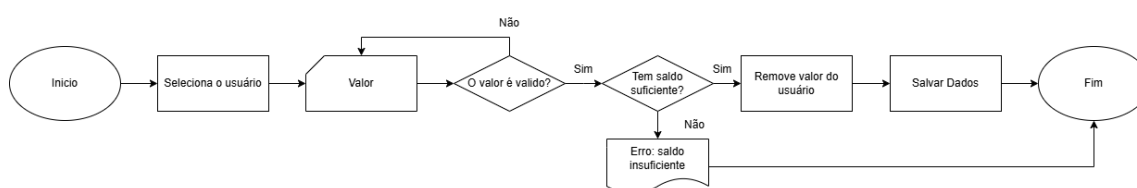



Saque de valor

Descrição

O saque de valor, deve selecionar um usuário por ID, usando a seleção de usuário, ler um valor superior a zero, que deverá ser removido de seu saldo. Caso o valor seja inválido, deve solicitar ao usuário digitá-lo novamente até ele inserir um valor válido. Caso o valor seja superior ao valor de seu saldo, deve exibir uma mensagem de erro e terminar a execução. Ao fim deve salvar os dados no arquivo externo.

Fluxograma





Pseudocódigo

```
Sacar_Valor

INICIO
  SE A tamanho(listaUsuarios) == 0 ENTAO
    ESCRIVER "Nao ha usuarios cadastrados!"
    RETORNAR NULL
  FIM SE
  DECLARAR indice
  indice <- selecionarUsuario()
  DECLARAR usuario
  usuario <- listaUsuarios[indice]
  DECLARAR valor
  ENQUANTO VERDADEIRO FACA
    ESCRIVER "Digite o valor do deposito:"
    LER valor
    SE valor <= 0 ENTAO
      ESCRIVER "Quantia inválida!"
      CONTINUAR
    FIMSE
  FIMENQUANTO
  SE valor > usuario.saldo ENTAO
    ESCRIVER "Valor de saque é superior ao saldo disponivel!"
  SENAO
    ESCRIVER "Saque realizado com sucesso!"
    usuario.saldo <- usuario.saldo - valor
  FIM SE
  SALVAR_DADOS_USUARIOS(listaUsuarios)
FIM
```



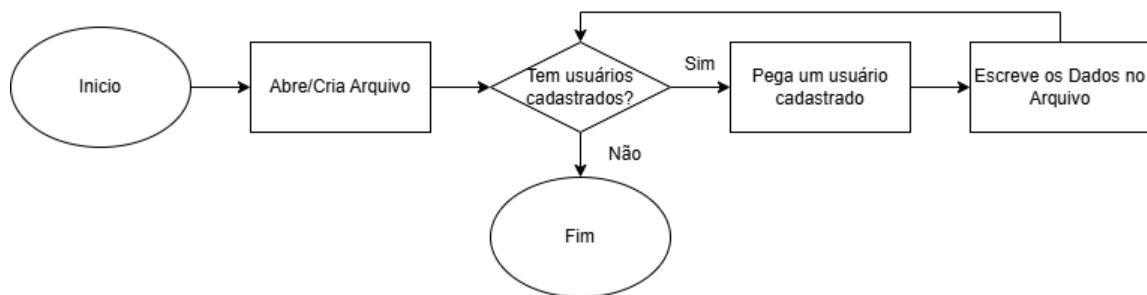
Salvar/Carregar Dados

Descrição

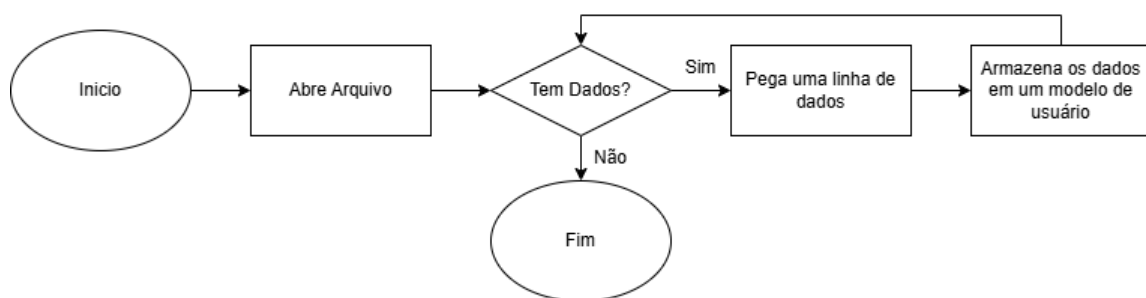
Salvar os dados do usuário deve escrever o nome, idade e saldo atual no arquivo externo, seguindo o modelo demonstrado nesse documento, caso não exista o documento deve cria-lo. Carregar os dados dos usuários deve ler os dados escritos no arquivo externo e armazena-lo na memória interna do software.

Obs.: Como esta funcionalidade é algo próprio de linguagem para linguagem, não há pseudocódigo.

Fluxograma Salvar Dados



Fluxograma Carregar Dados





Observações

Programa desenvolvido com o intuito educativo, para a disciplina de Laboratório de Introdução a Ciências da Computação (SSC0502) e Introdução a Ciências da Computação (SSC0501) do curso de Bacharelado em Sistemas da Informação ofertada pelo ICMC da USP de São Carlos.

Para melhor administrar os dados do usuário deve se realizar uma biblioteca de vector autoral, para realizar procedimentos relacionados de forma mais simples e prática sistemas como alocação dinâmica, adição e remoção de usuários e/ou outros itens. Além de uma biblioteca de controle de arquivos e de usuário.

Sistema Operacional Utilizado Para o Desenvolvimento: *Windows 11 x64*

Software de Desenvolvimento: *CLion – Plano Gratuito sem intuito corporativo*

Linguagem de Desenvolvimento: *C*

O programa deve ser salvo em forma visionada utilizando Git, e postado de forma publica no Github. Acesso ao código em <https://github.com/ArthurAzorli/UaiBank>