

Implementação Sistema de Arquivos

Leônidas Serra, Isabela Oliveira, Paulo Arthur, Samuel Basílio

Objetivos

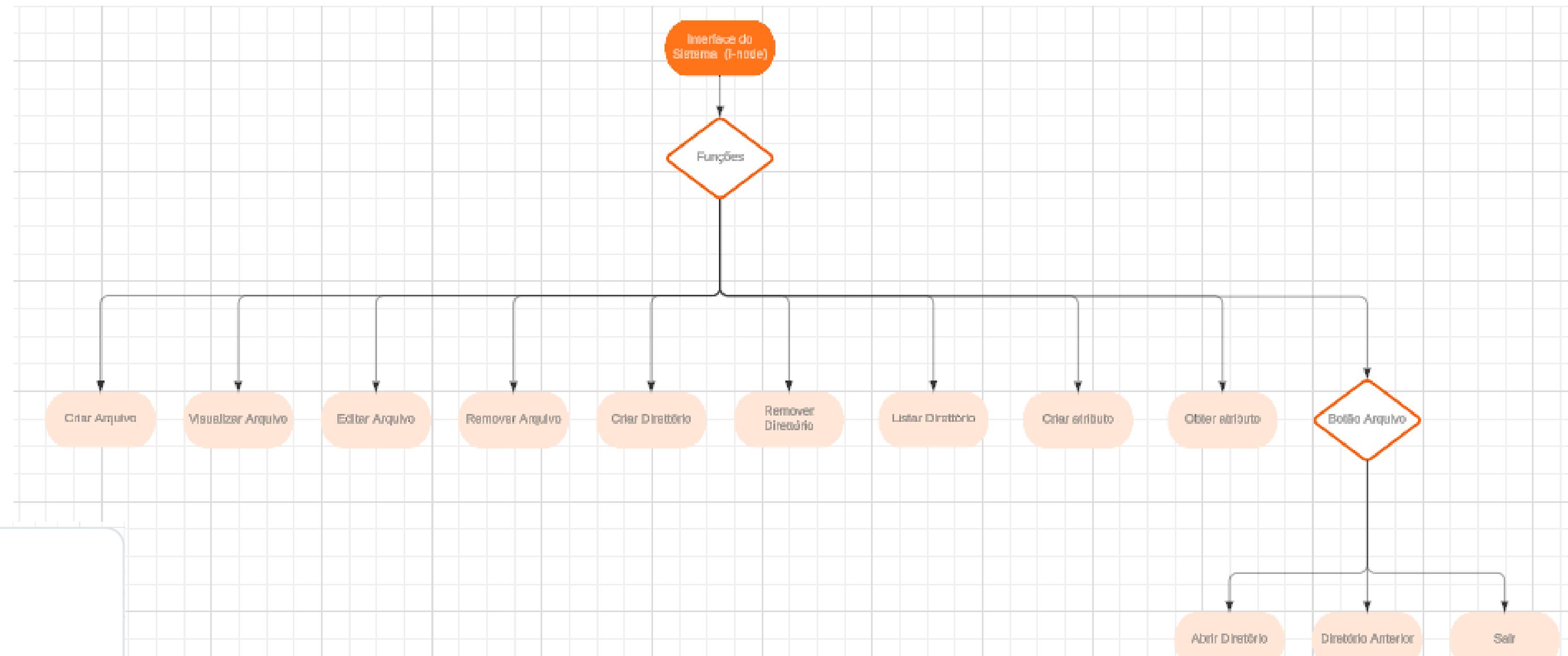
Pontos para discussão:

- Apresentar a implementação de Sistemas de Arquivos baseados em I-node e em Tabela de Blocos Livres
- Fazer um comparativo dos recursos entre ambos os modelos de Sistemas de Arquivos

Referências

- Os sistemas são baseados em recursos não implementados na disciplina até então
- Documentação biblioteca Tkinter

Interface I-node



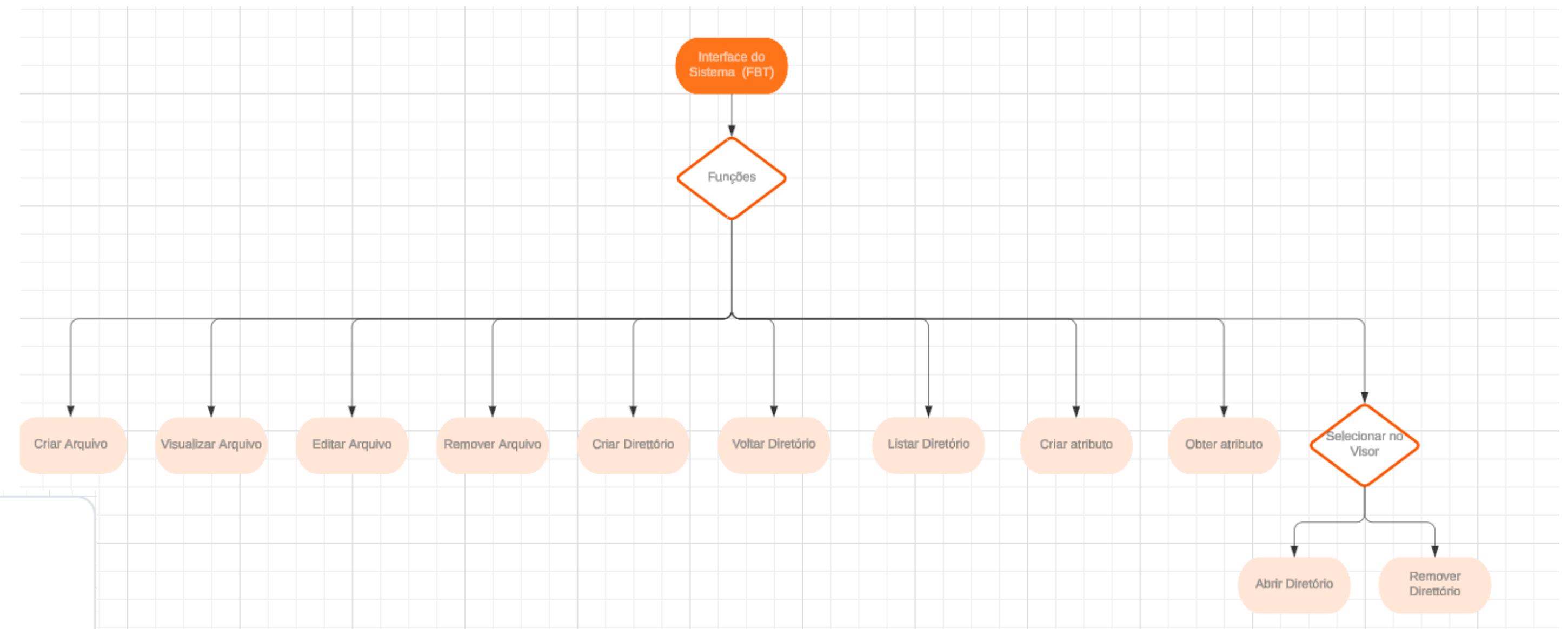
Legenda do fluxorama

Contexto

Recurso

Funcionalidades

Interface Tabela de blocos



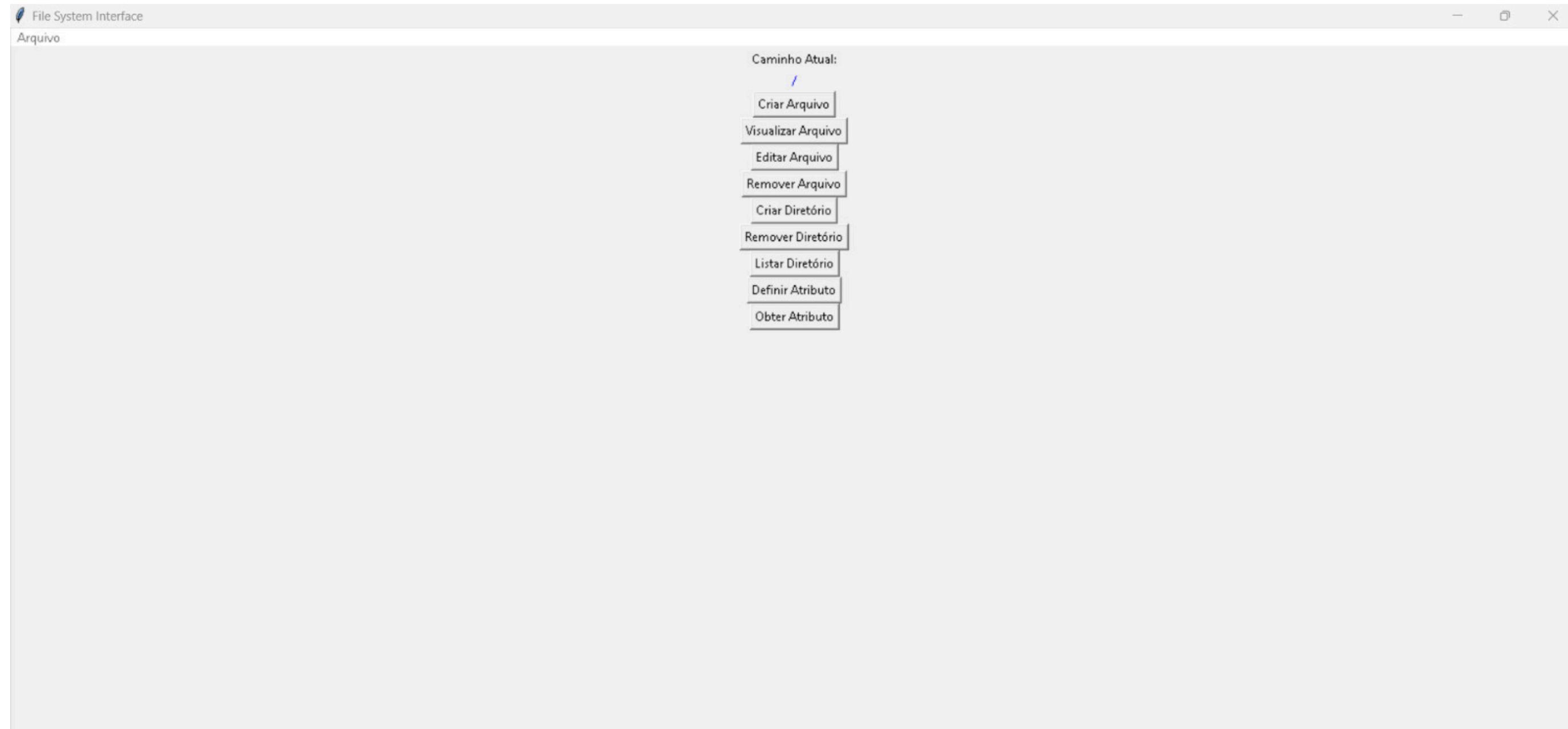
Legenda do fluxorama

Contexto

Recurso

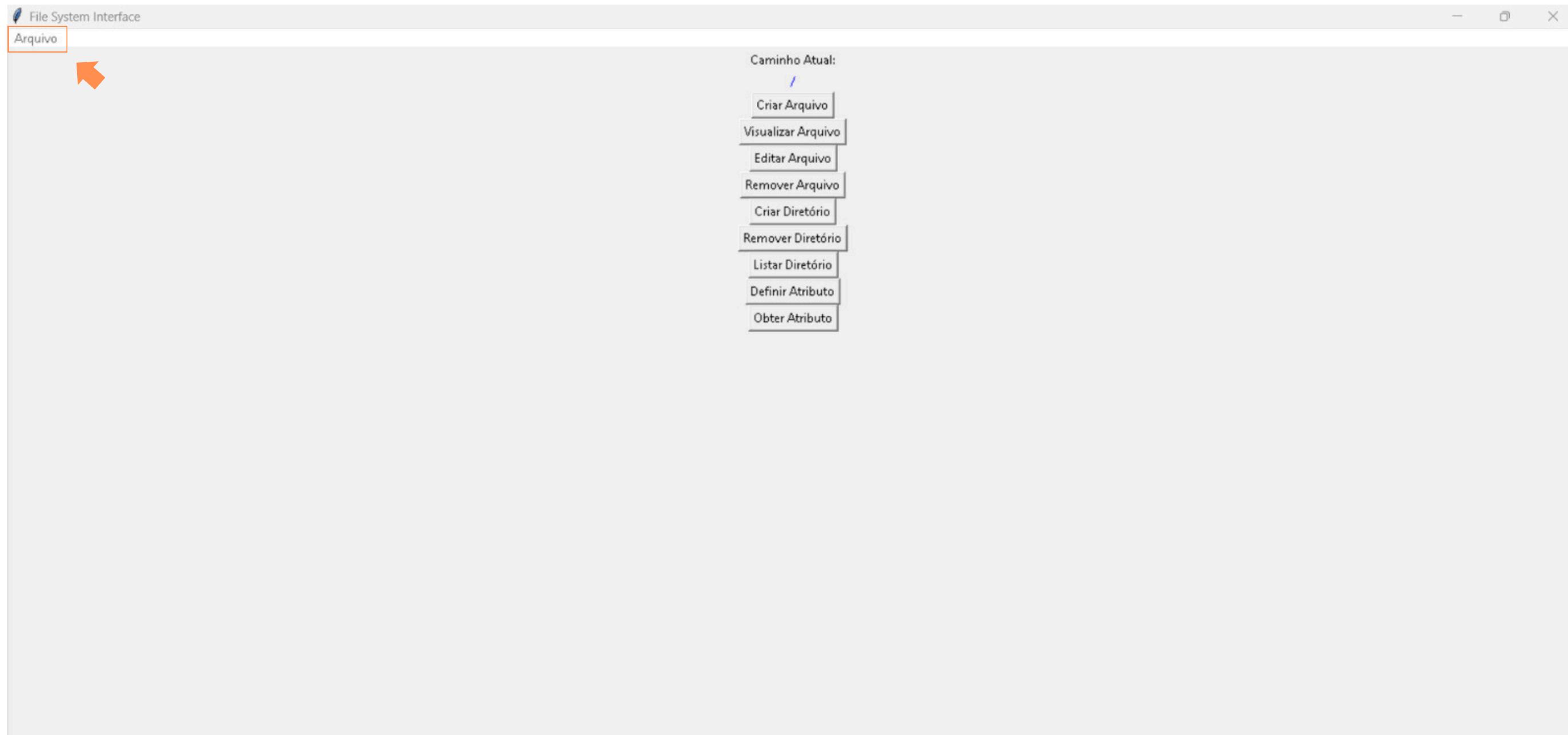
Funcionalidades

Comparativo entre interfaces



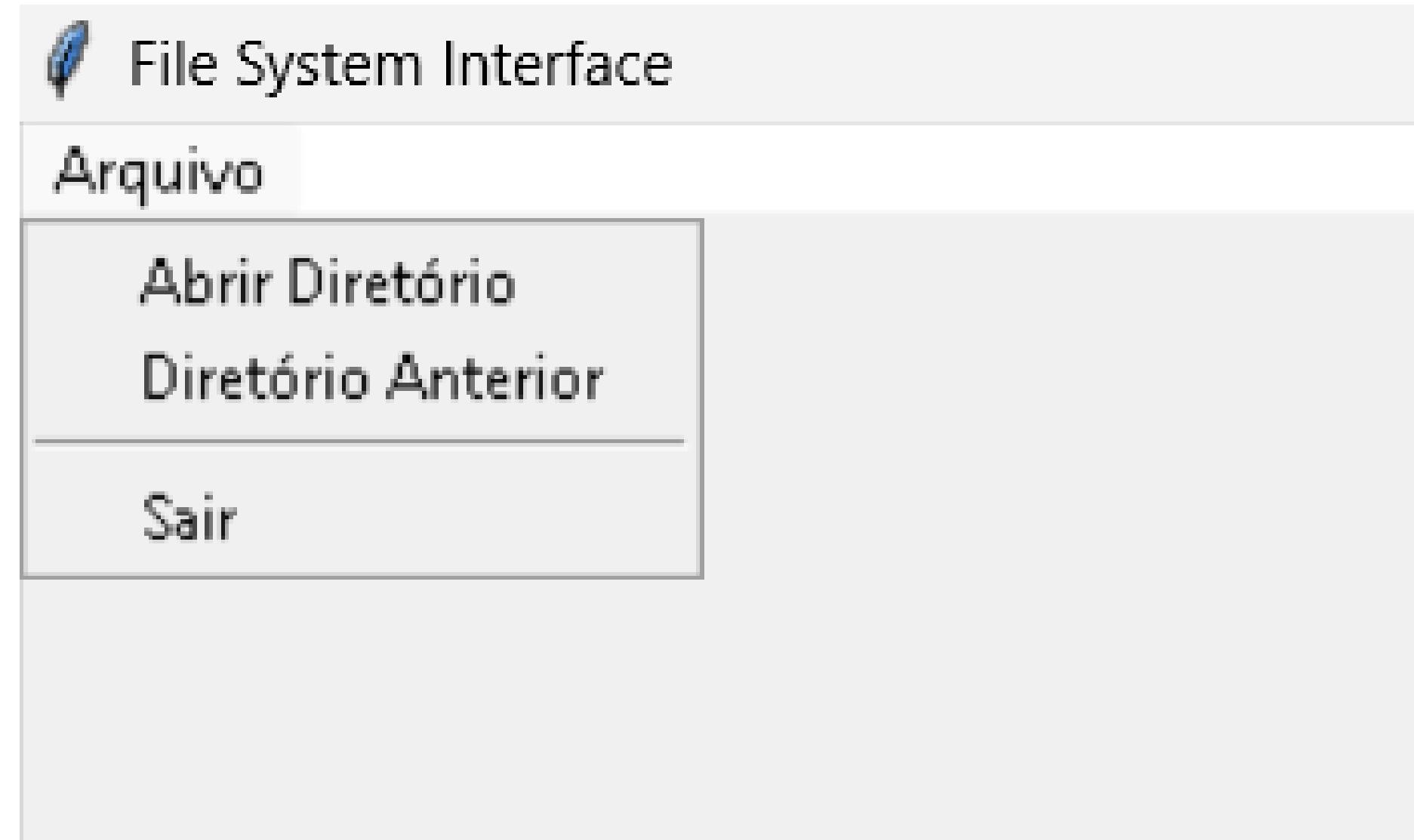
Em I-node

Comparativo entre interfaces



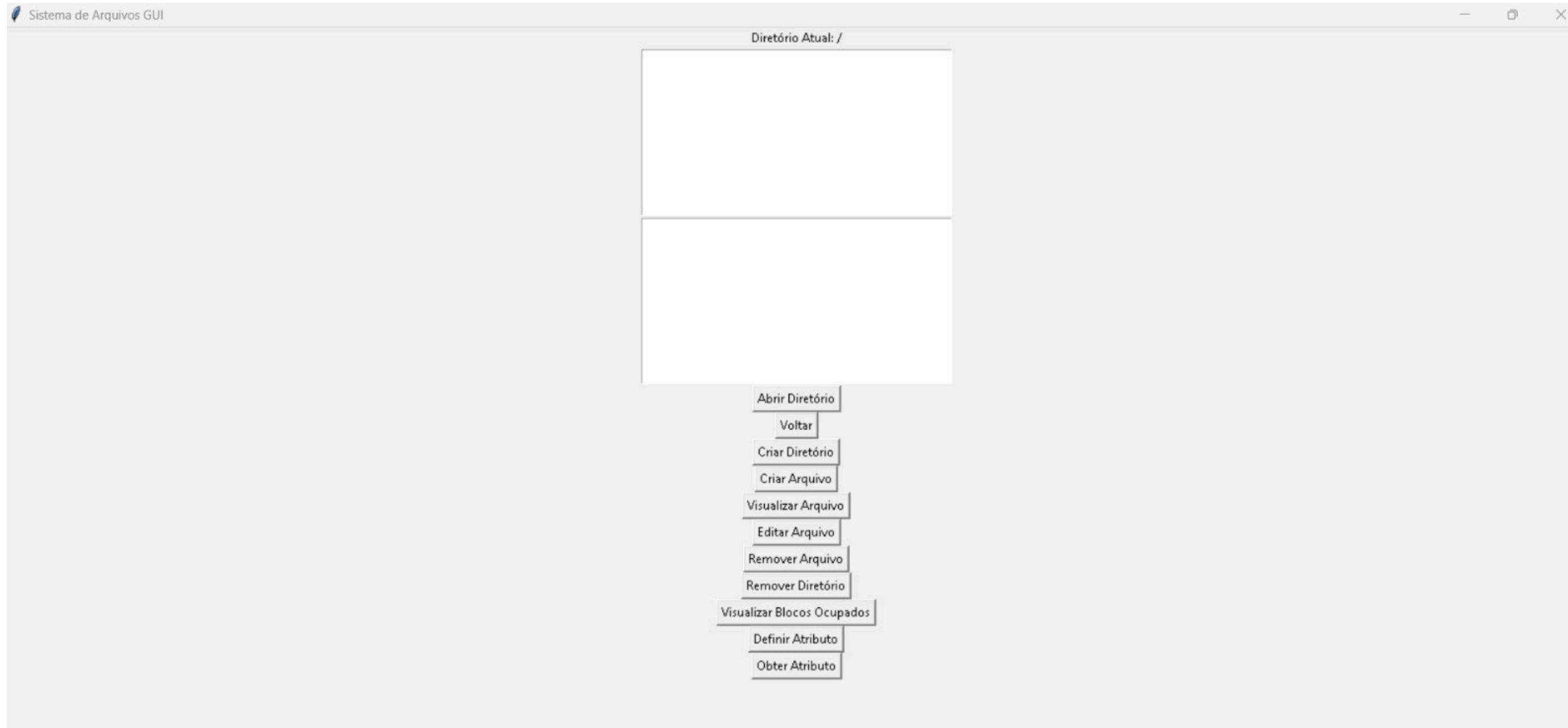
Em I-node

Comparativo entre interfaces



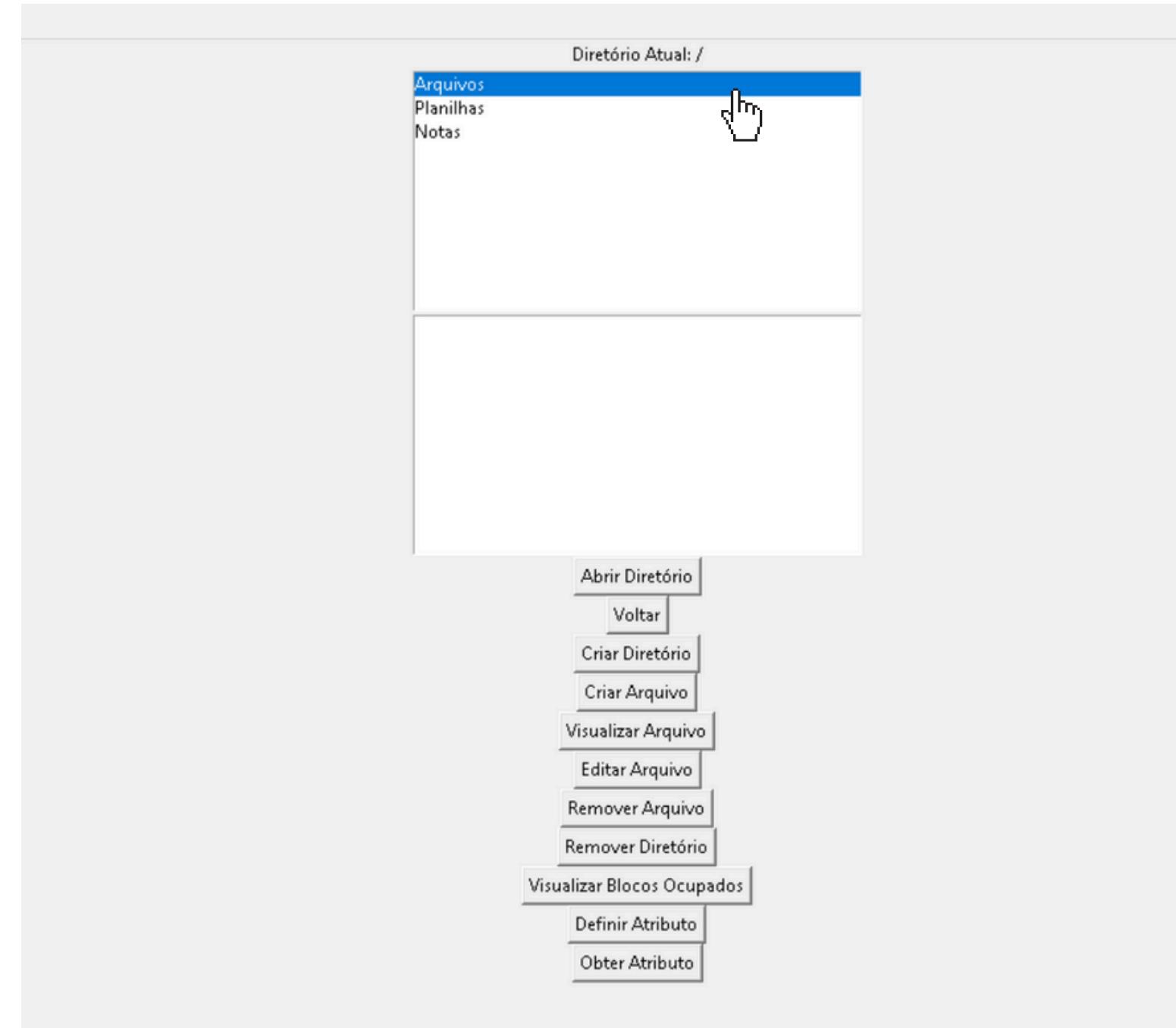
Em I-node

Comparativo entre interfaces



Em tabela de blocos livres

Comparativo entre interfaces

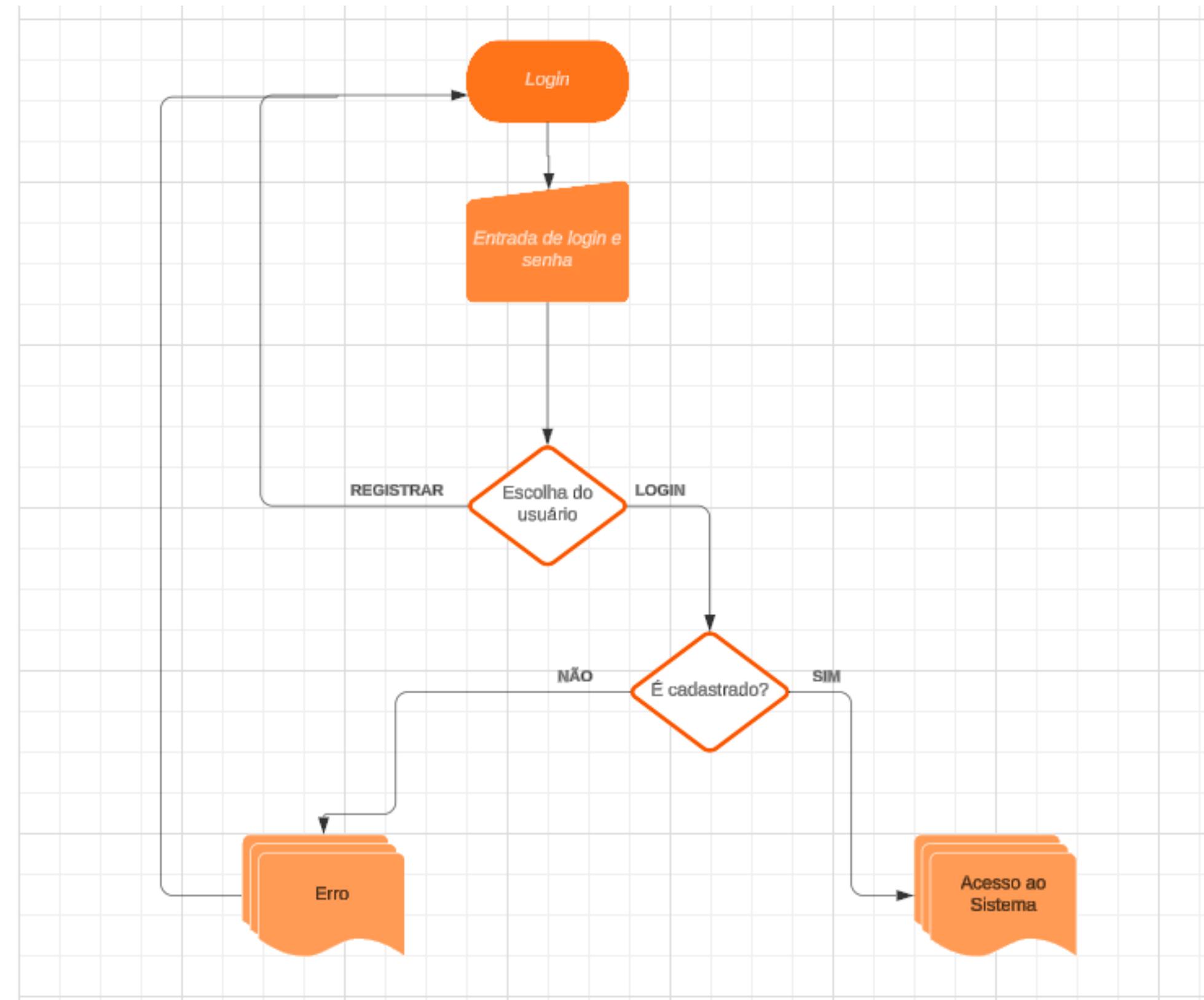


Em tabela de blocos livres

Login

- 01 O método de login é igual para ambos os sistemas**
- 02 Os usuários são guardados em um dicionário, no qual a chave e o valor são strings**
- 03 O login é definido pela classe LoginApp**

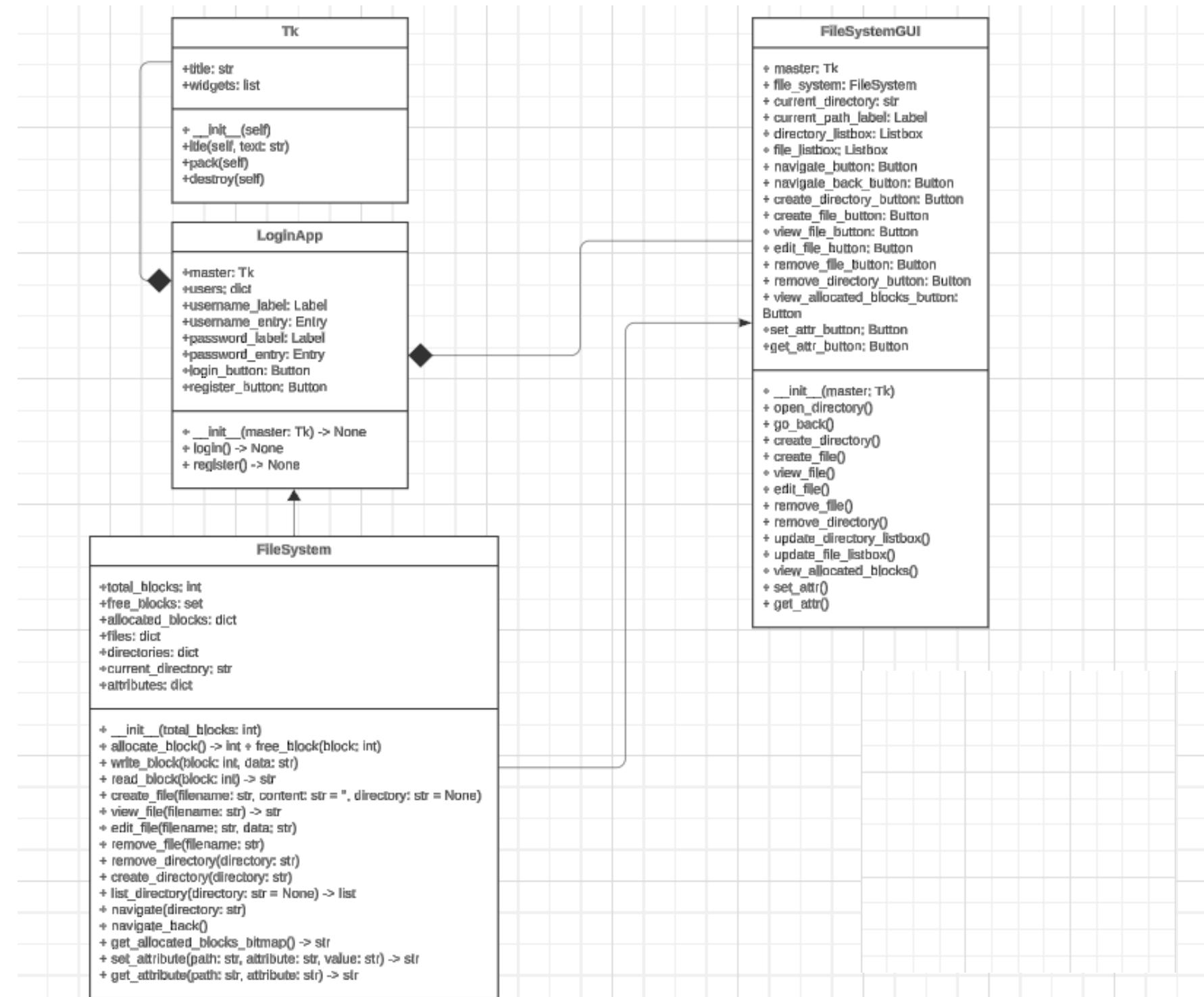
Login



Legenda do fluxograma

- Contexto
- Entrada do usuário
- Decisão
- Funcionalidades
- Retorno

Diagrama de Classes(Visão Geral)



Legenda do
diagrama

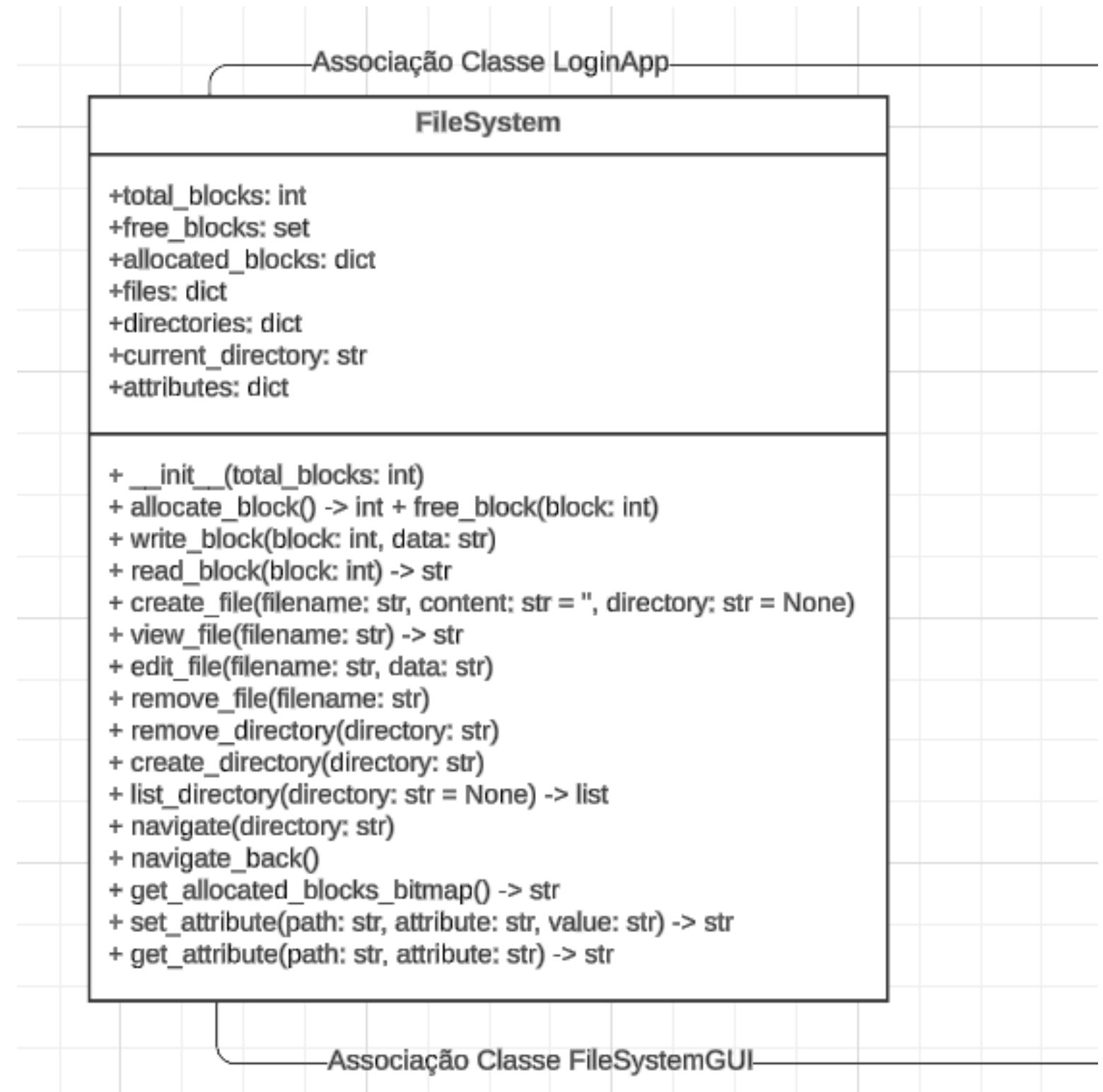
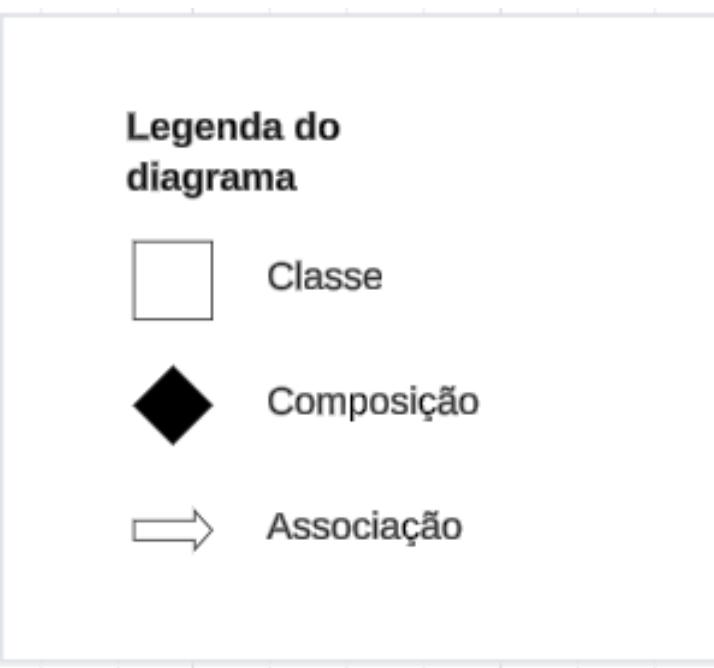
Classe

Composição

Associação

Para tabela de blocos livres

Diagrama de Classes



Para tabela de blocos livres

Diagrama de Classes

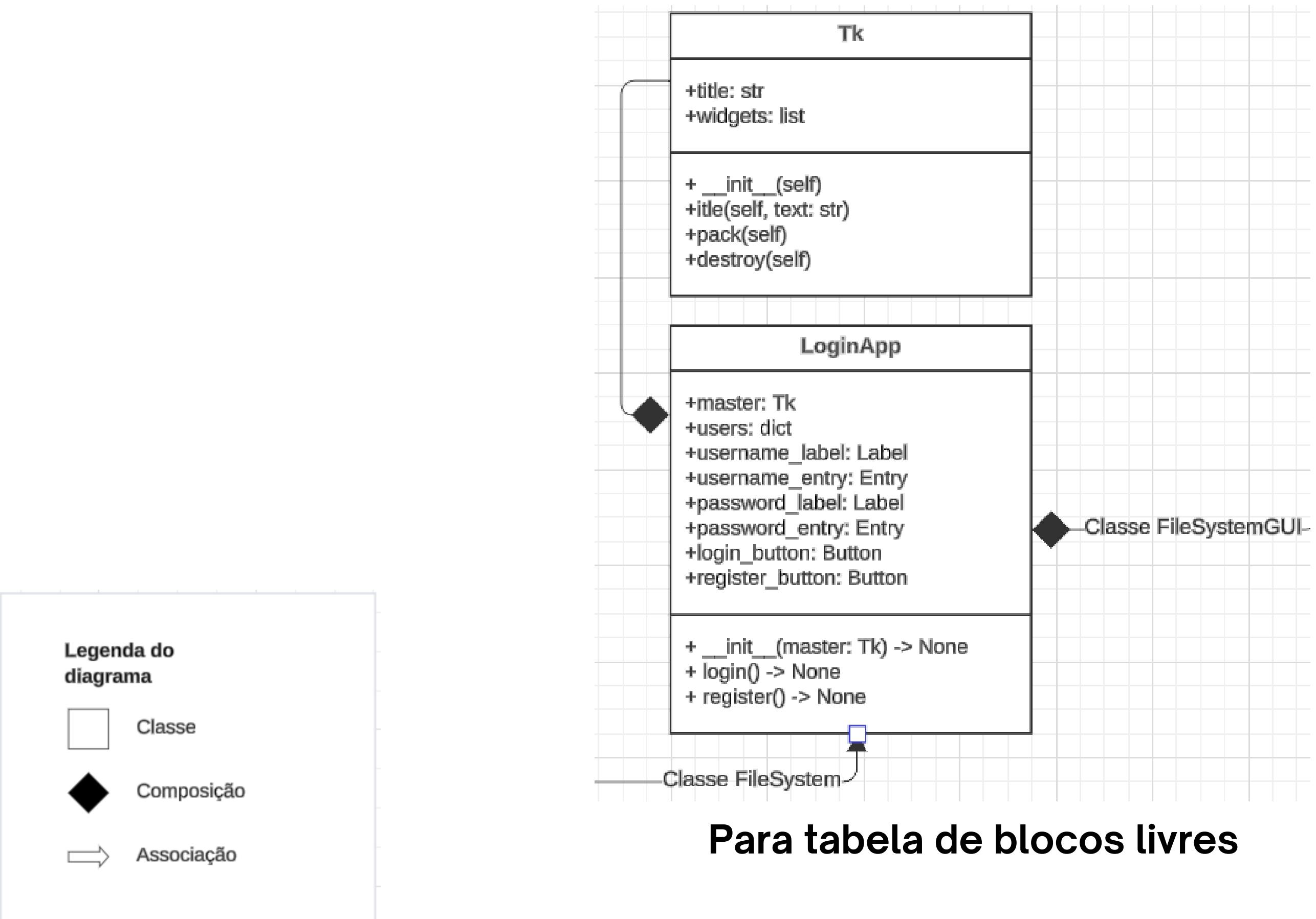
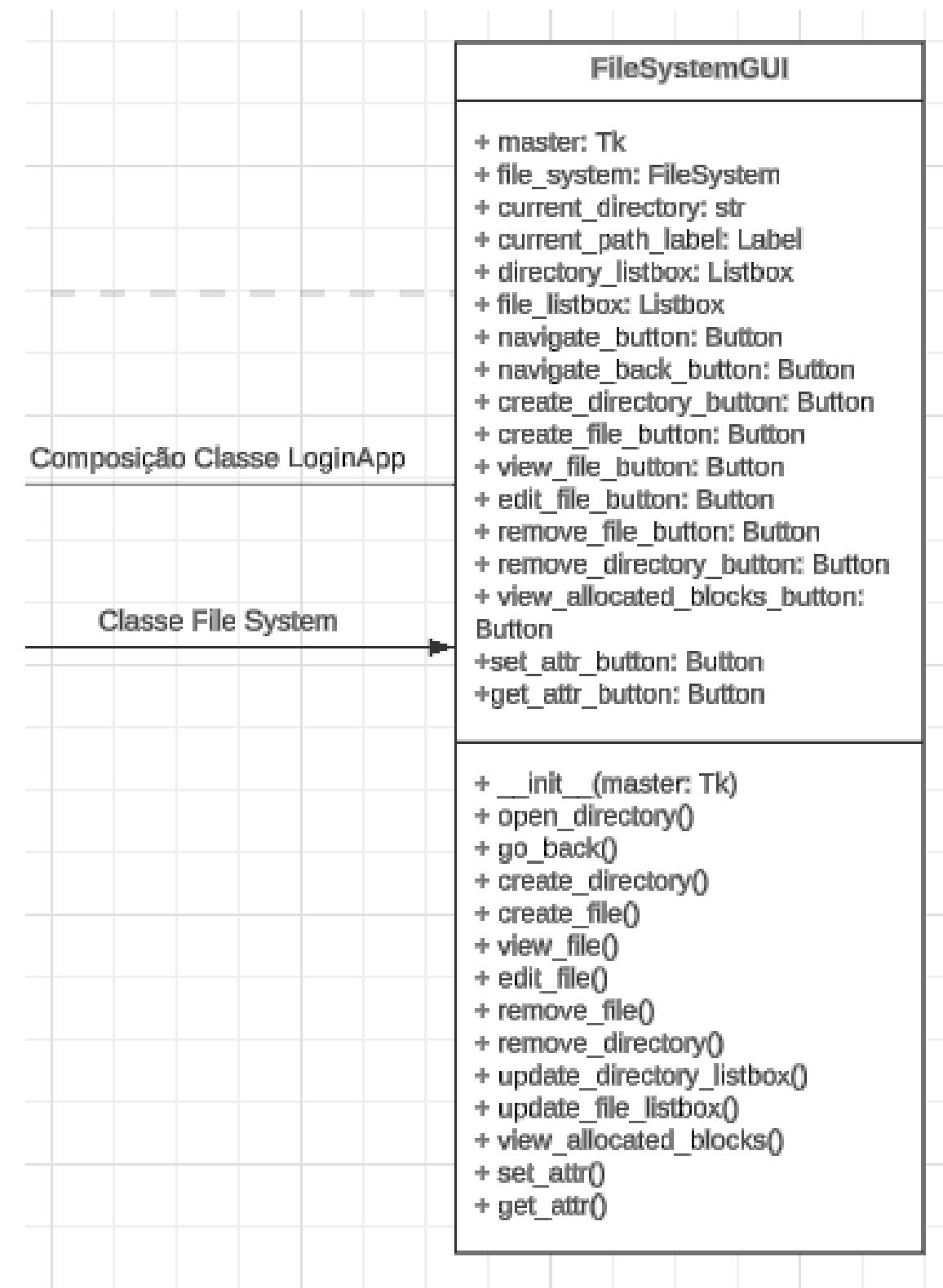
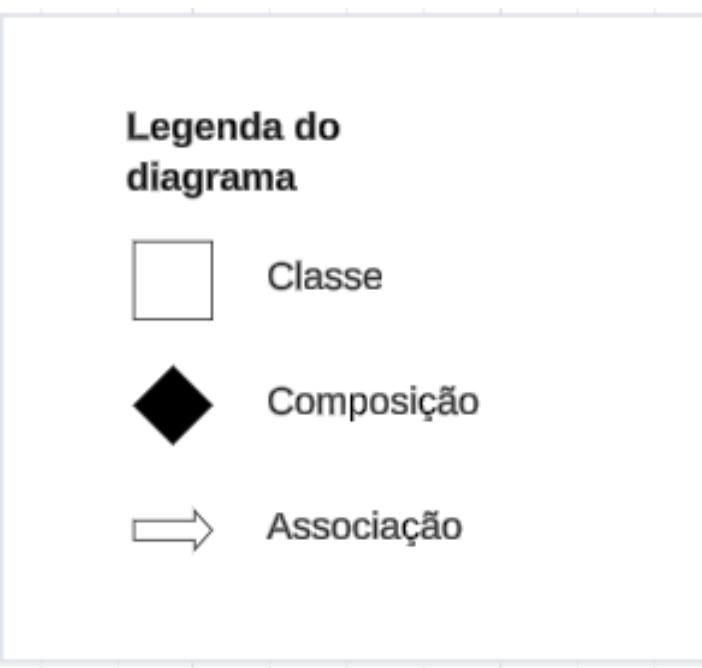
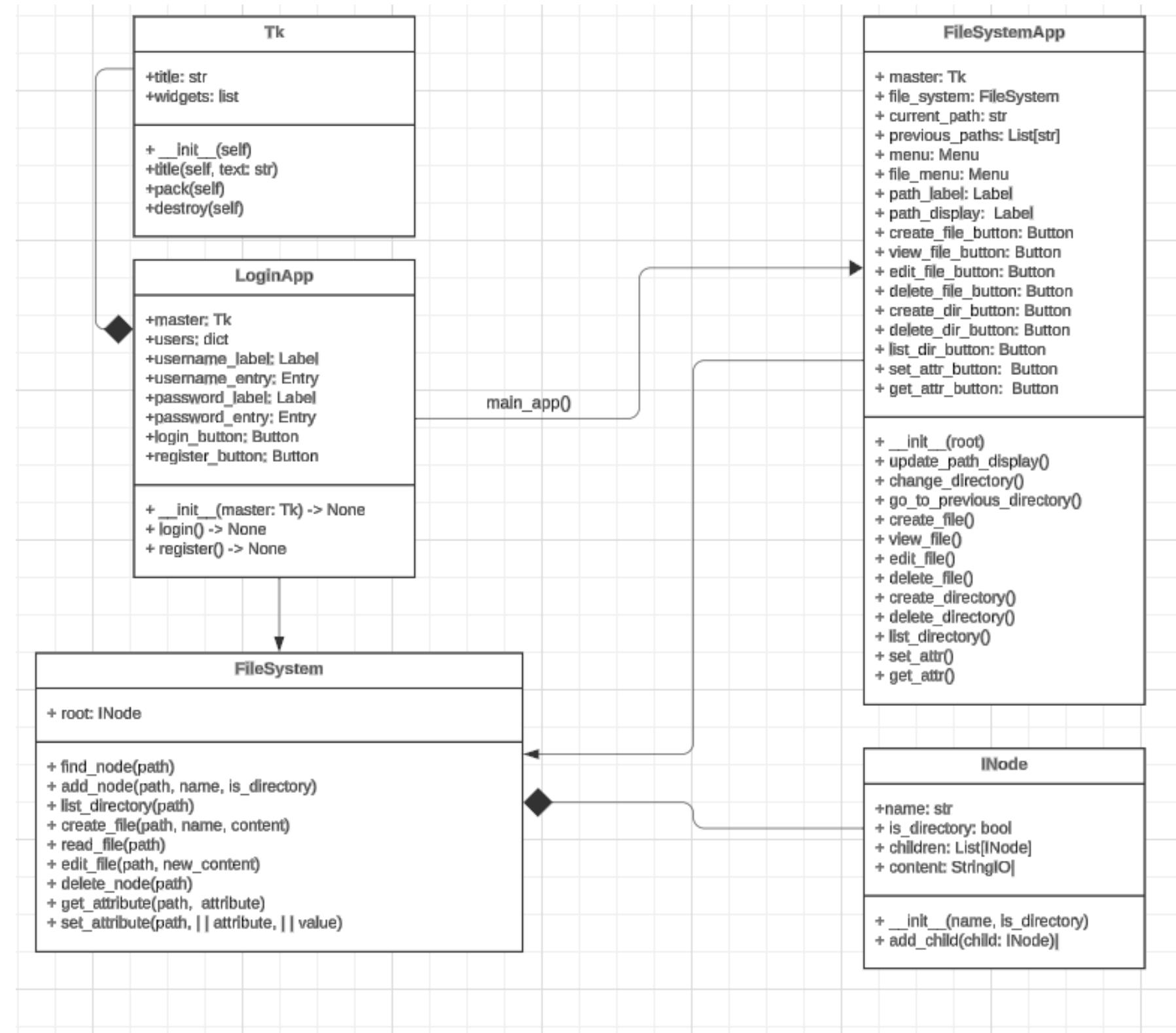


Diagrama de Classes



Para tabela de blocos livres

Diagrama de Classes(Visão Geral)



Legenda do
diagrama

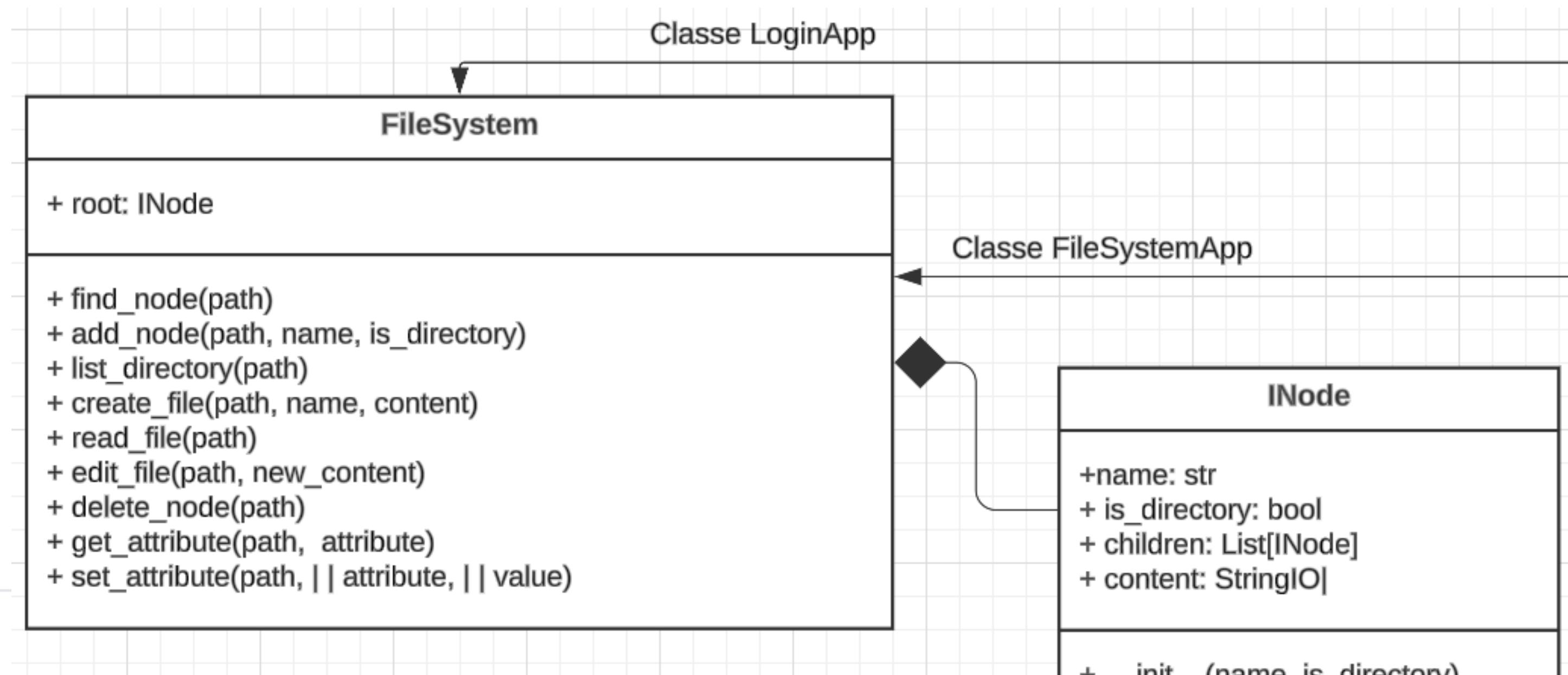
Classe

Composição

Associação

Para i-node

Diagrama de Classes



Legenda do
diagrama

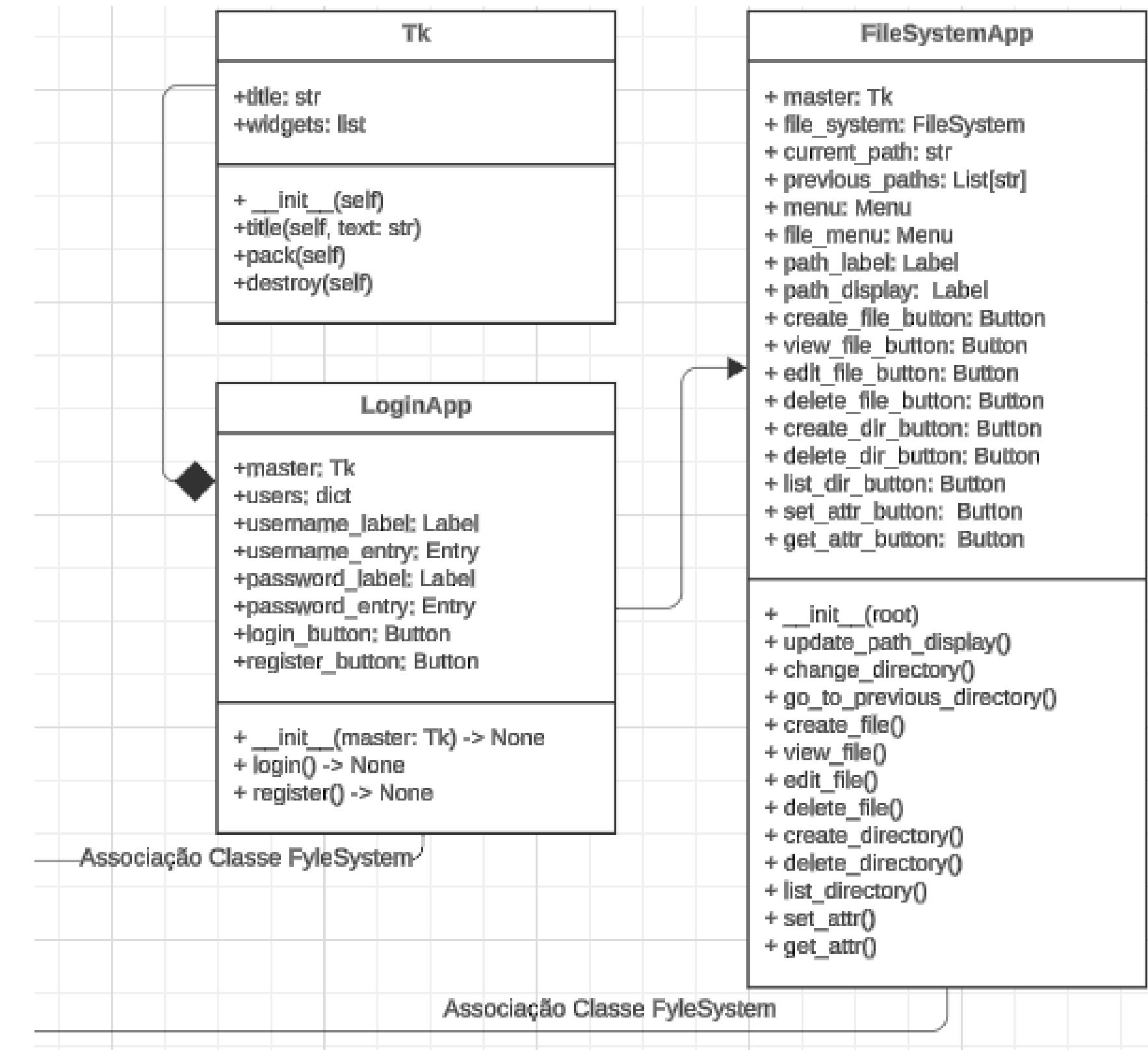
Classe

Composição

Associação

Para i-node

Diagrama de Classes

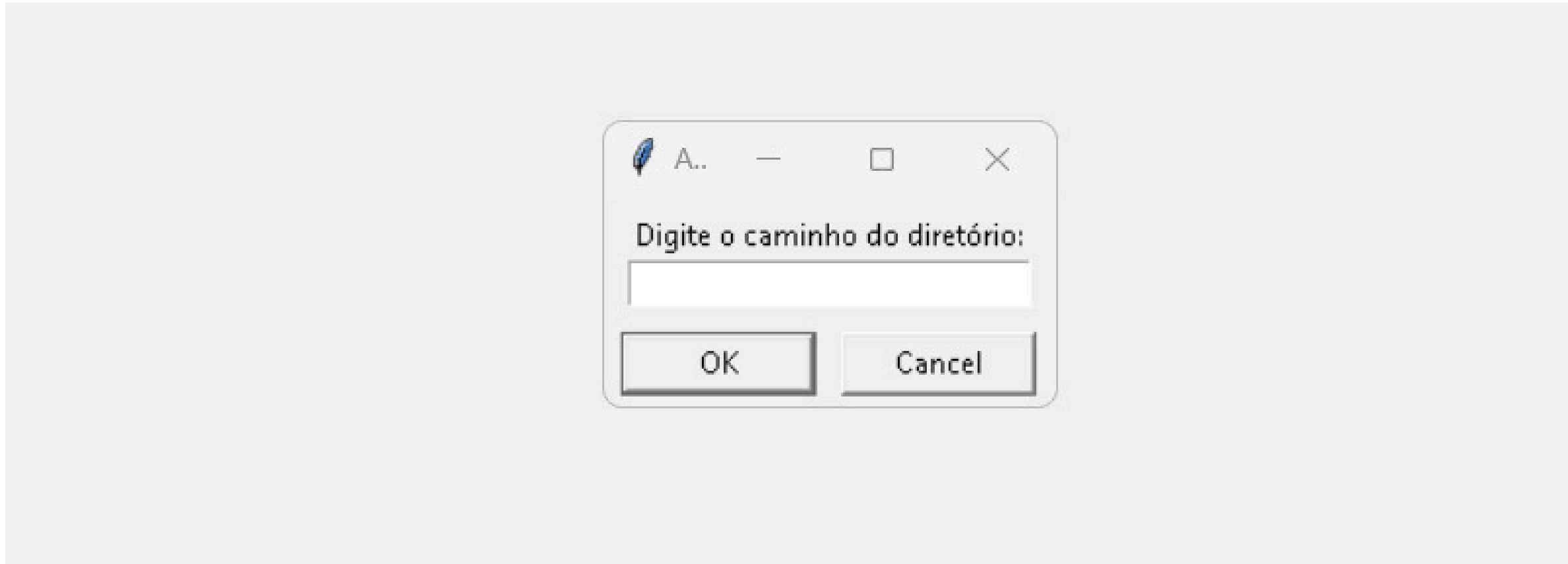


Para i-node

Funções da Classe de interface

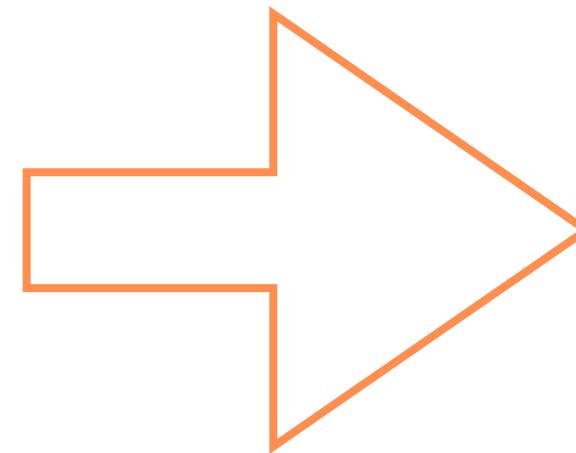
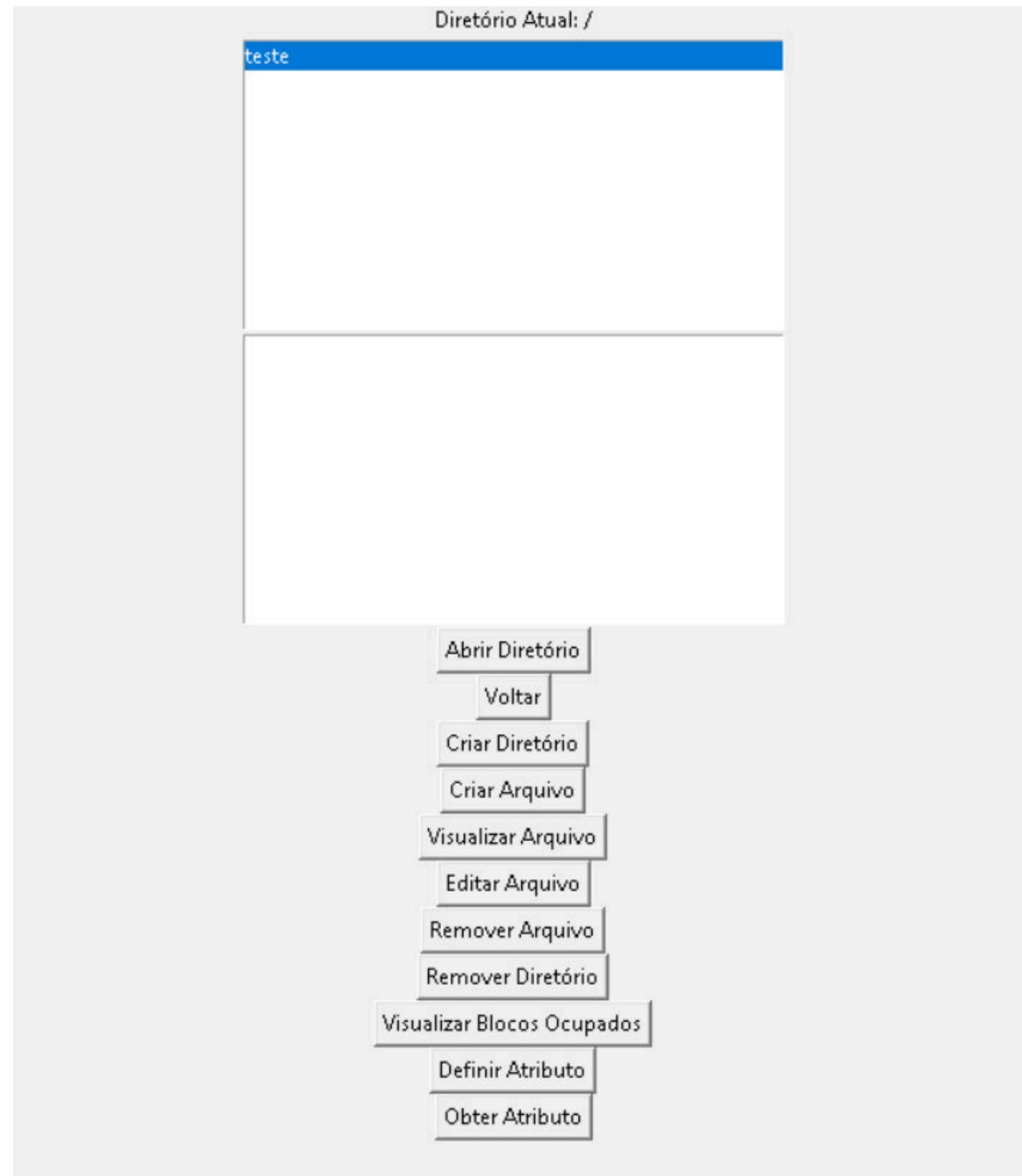
- 01 As funções da classe de interface fazem os mesmos passos das funções da Classe File_System**
- 02 As funções da interface utilizam o rótulo como maneira de armazenar o caminho do conteúdo a ser mostrado**
- 03 A principal diferença entre as funções das interface de i-node e tabela de blocos livres é o input**

Diferença de input



Em i-node

Diferença de input

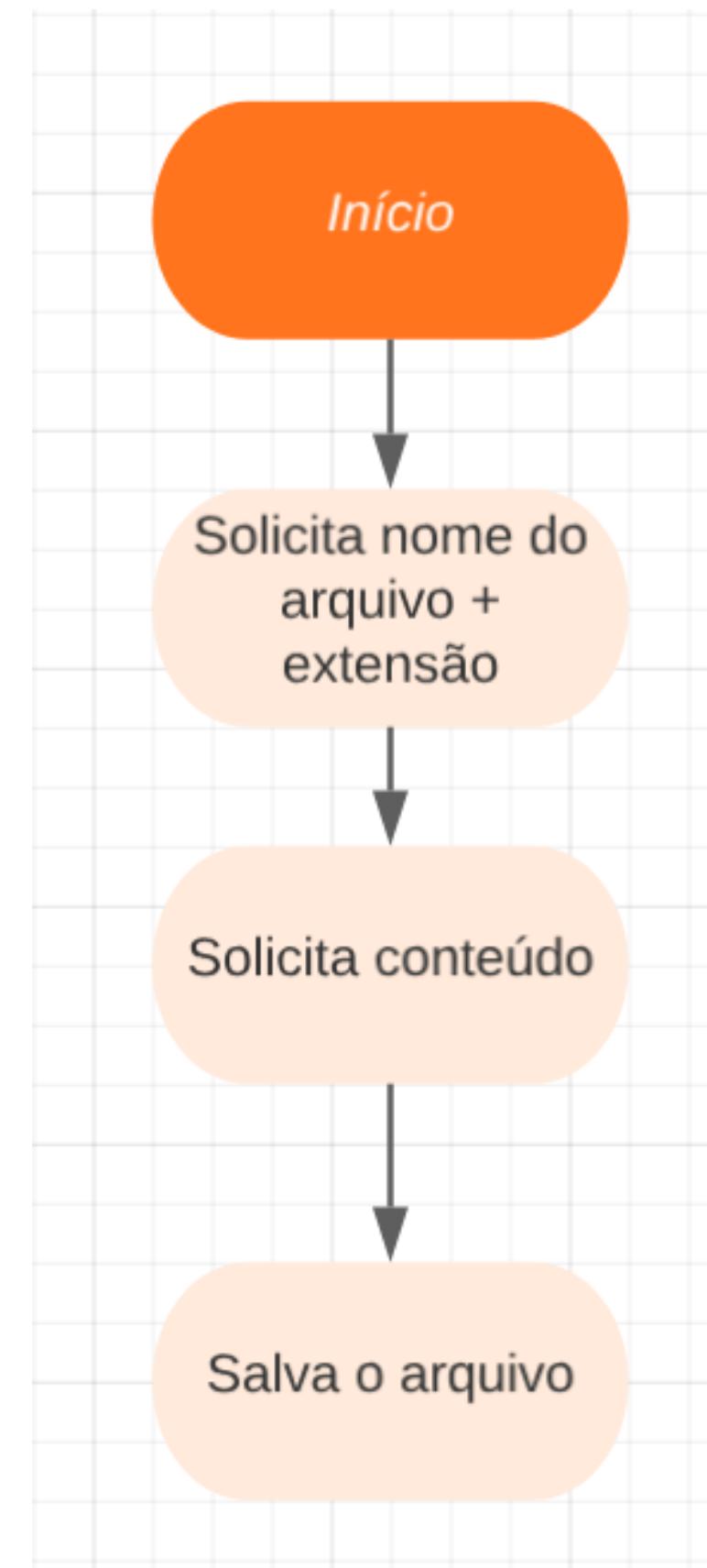


Em tabela de blocos livres

Função Criar arquivo I-Node

Legenda do fluxograma

- Contexto
- Decisão
- Funcionalidades



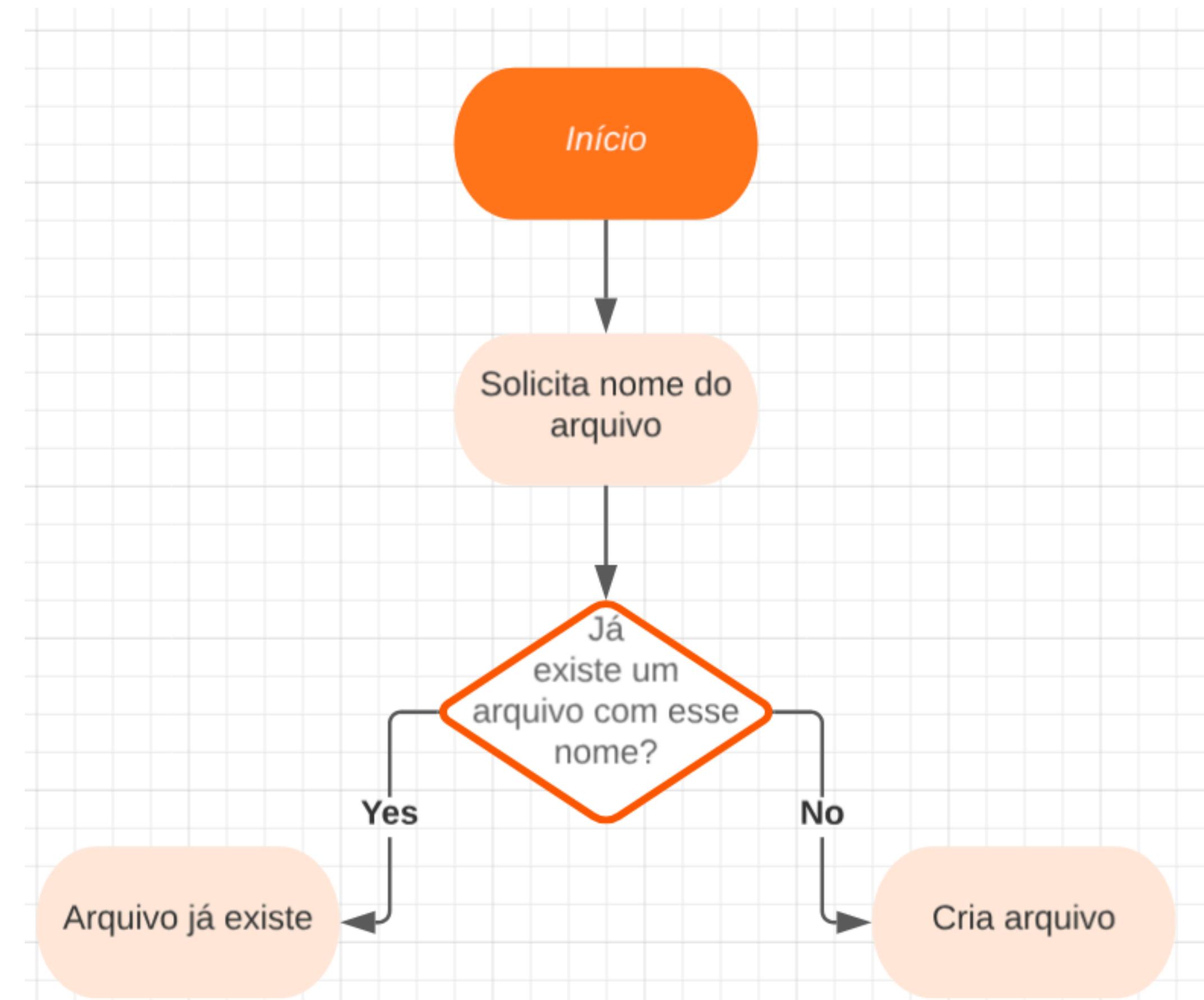
Função Criar arquivo - tabela de blocos livres

Legenda do fluxorama

Contexto

Decisão

Funcionalidades



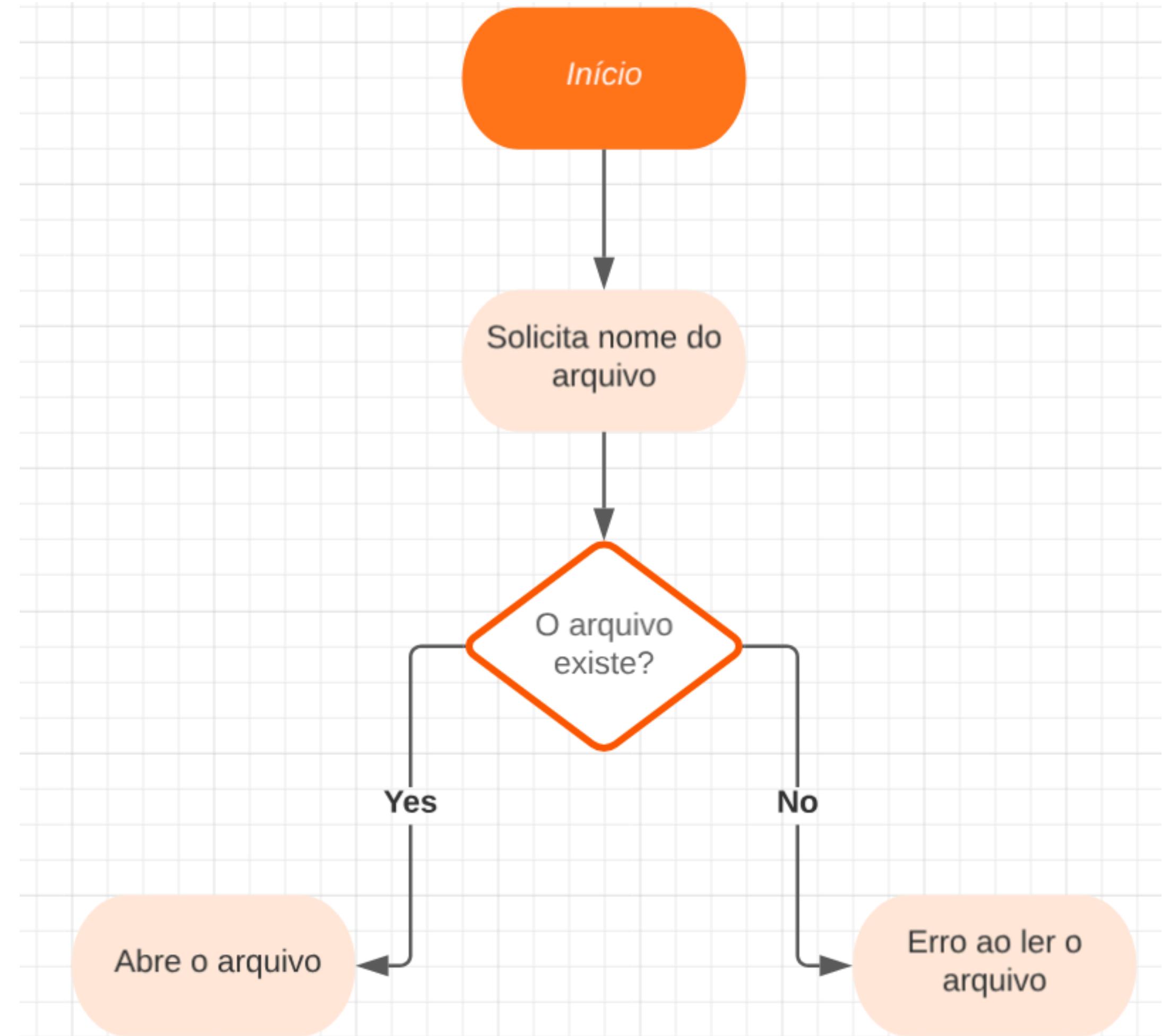
Função Visualizar arquivo I-Node

Legenda do fluxorama

Contexto

Decisão

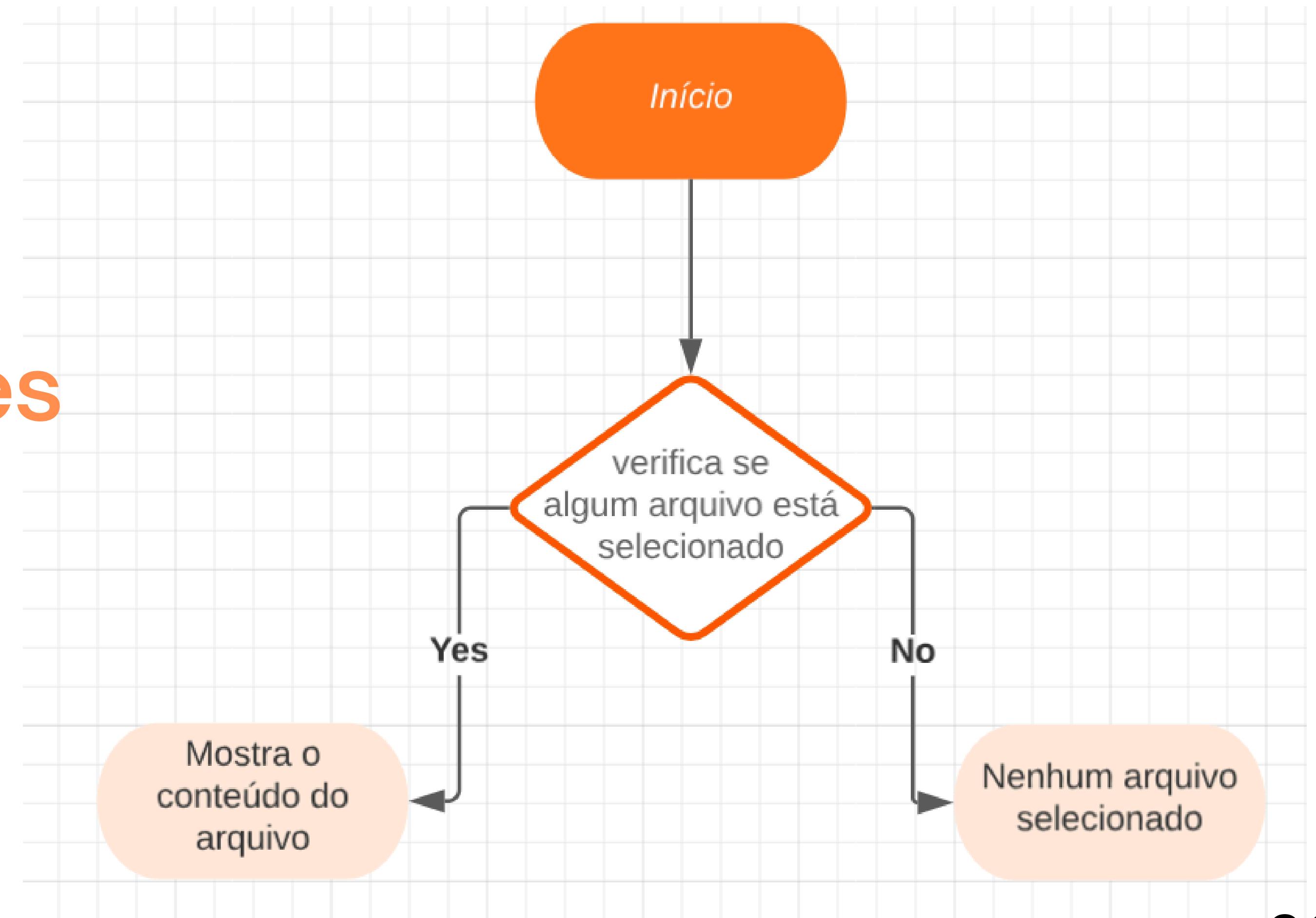
Funcionalidades



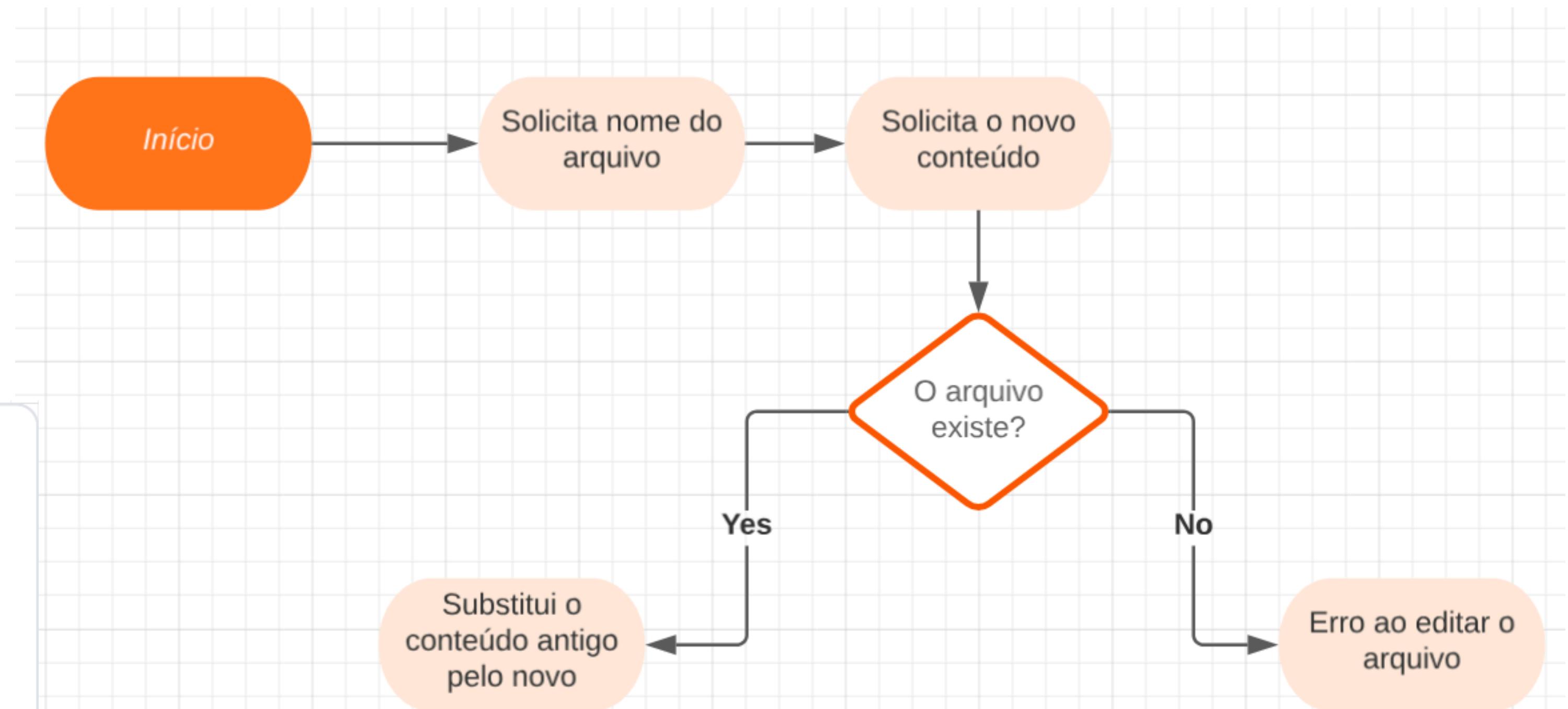
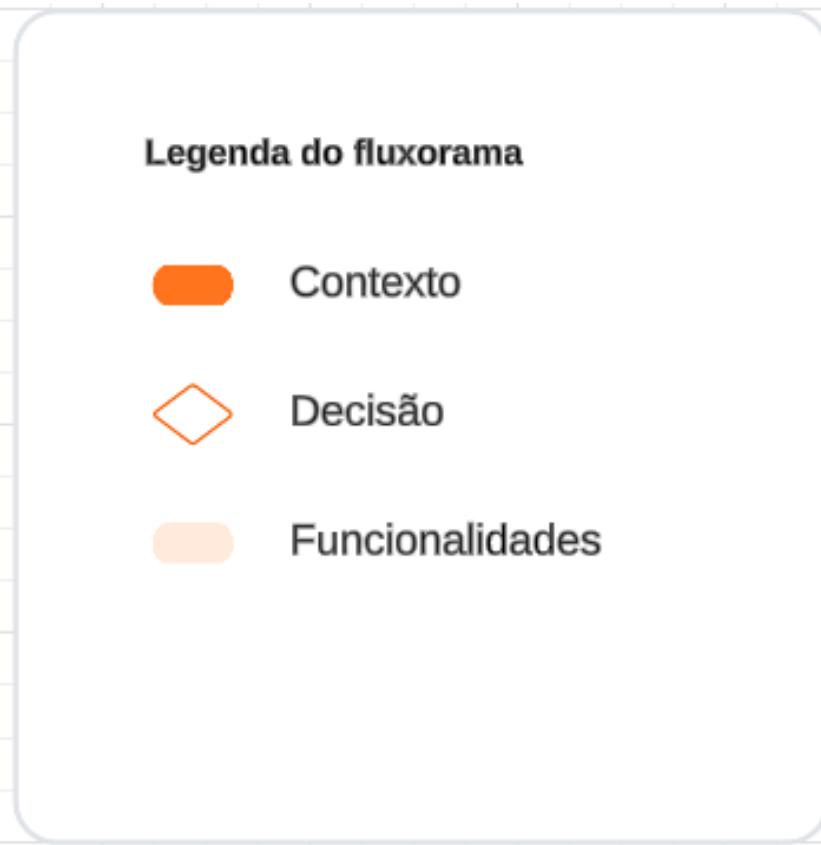
Função Visualizar arquivo - tabela de blocos livres

Legenda do fluxorama

- Contexto
- Decisão
- Funcionalidades



Função Editar arquivo I-Node



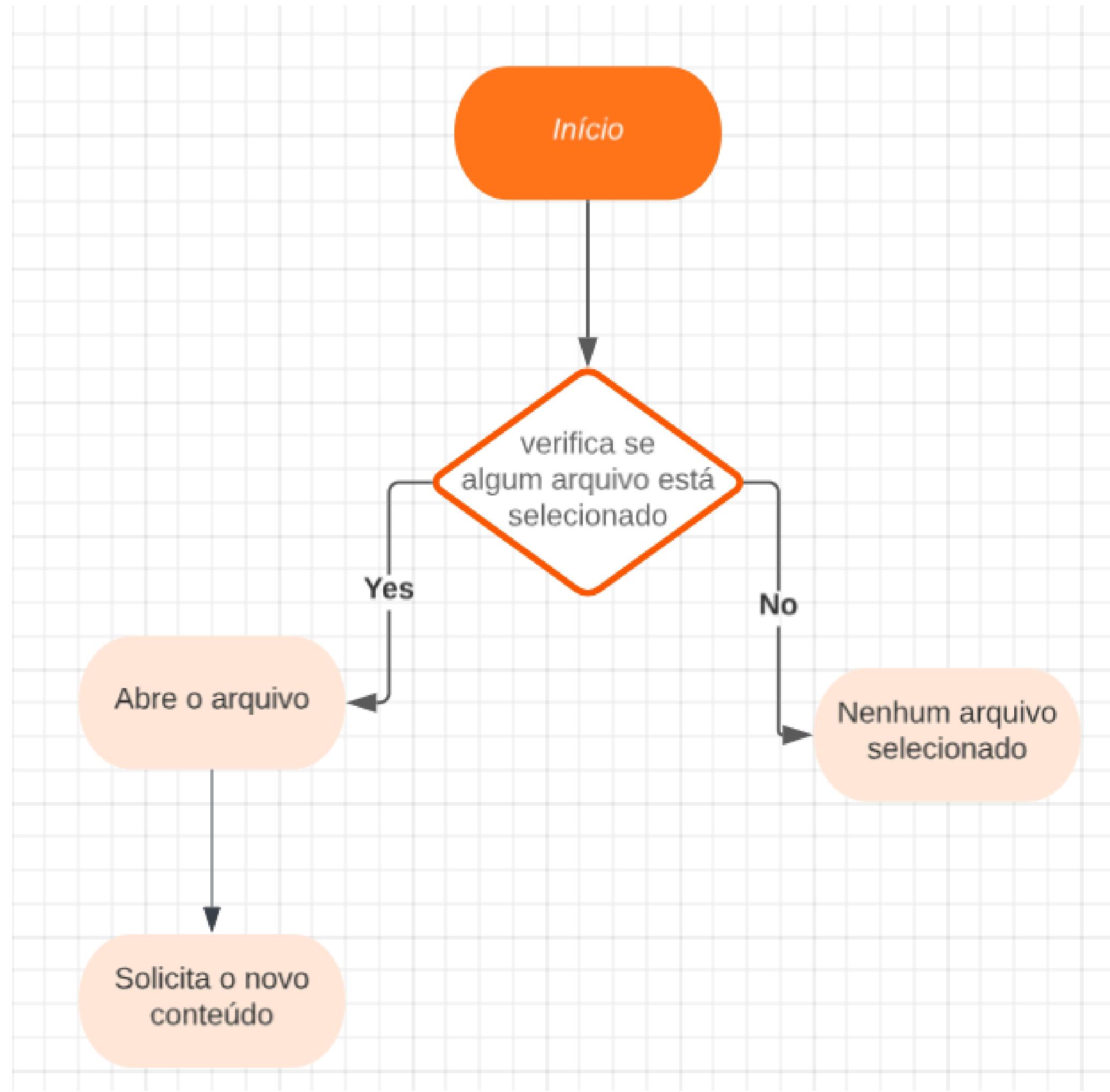
Função Editar arquivo- tabela de blocos livres

Legenda do fluxorama

Contexto

Decisão

Funcionalidades



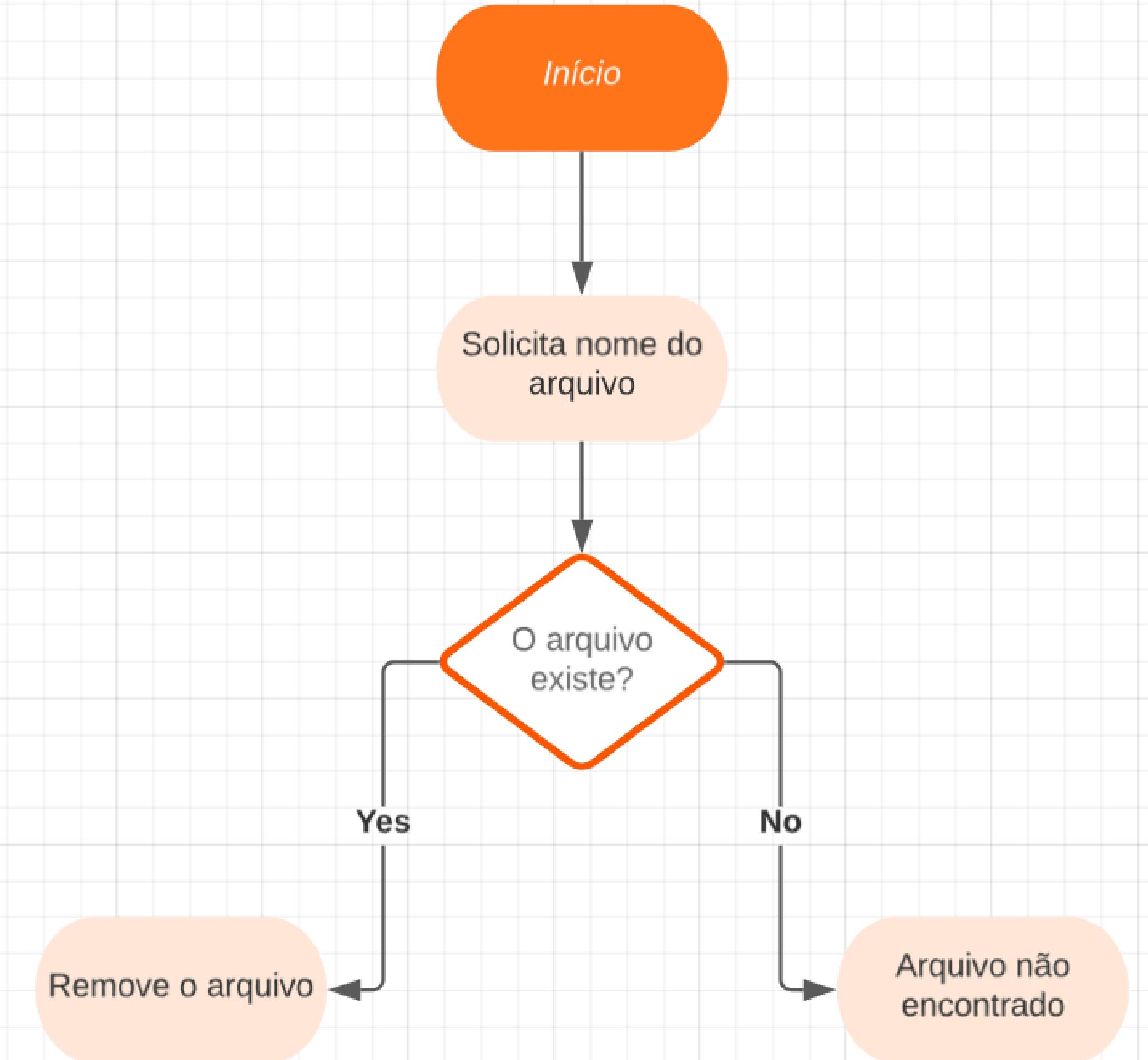
Função Remover arquivo - I-Node

Legenda do fluxorama

Contexto

Decisão

Funcionalidades



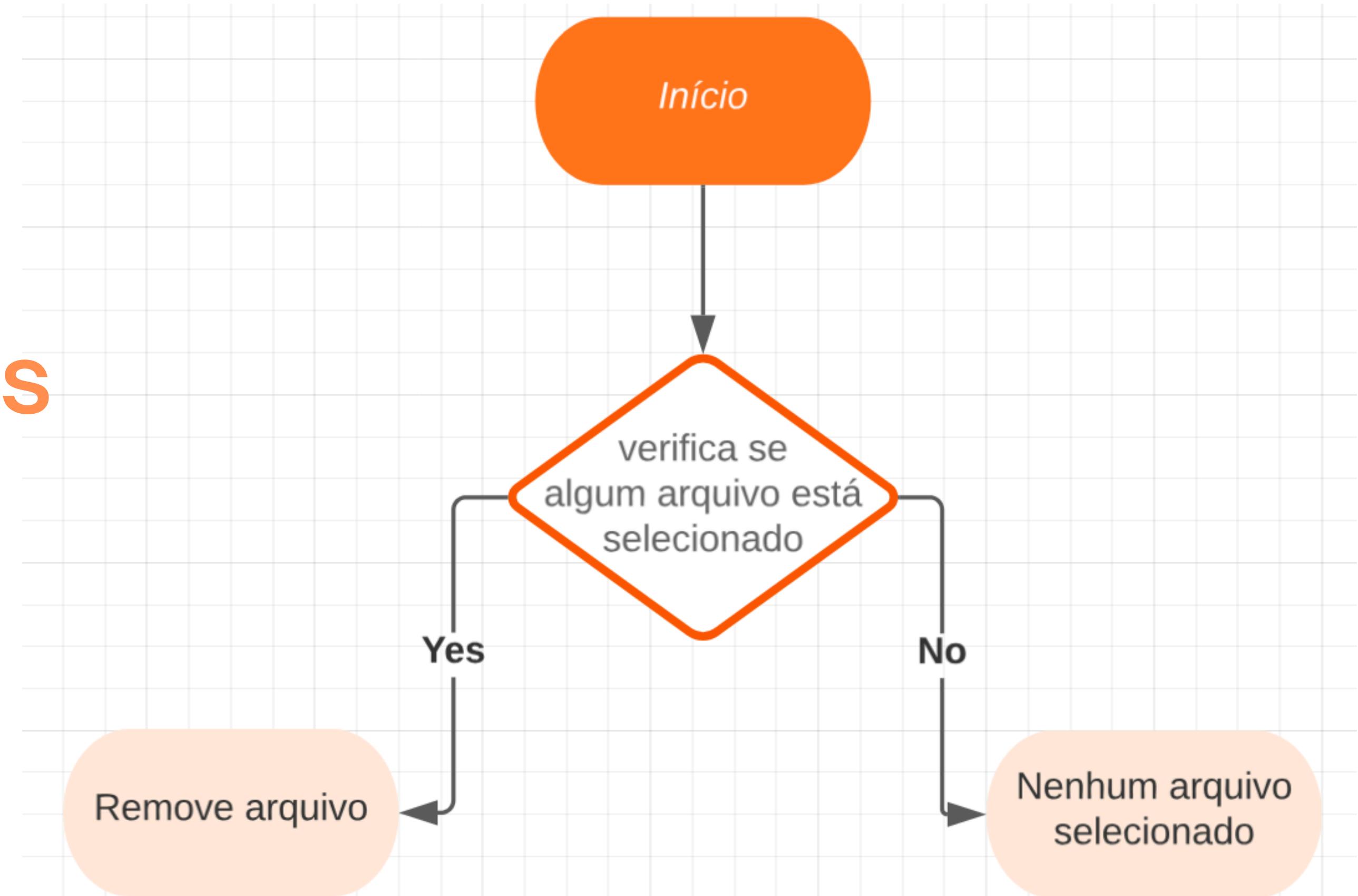
Função Remover arquivo - tabela de blocos livres

Legenda do fluxorama

Contexto

Decisão

Funcionalidades



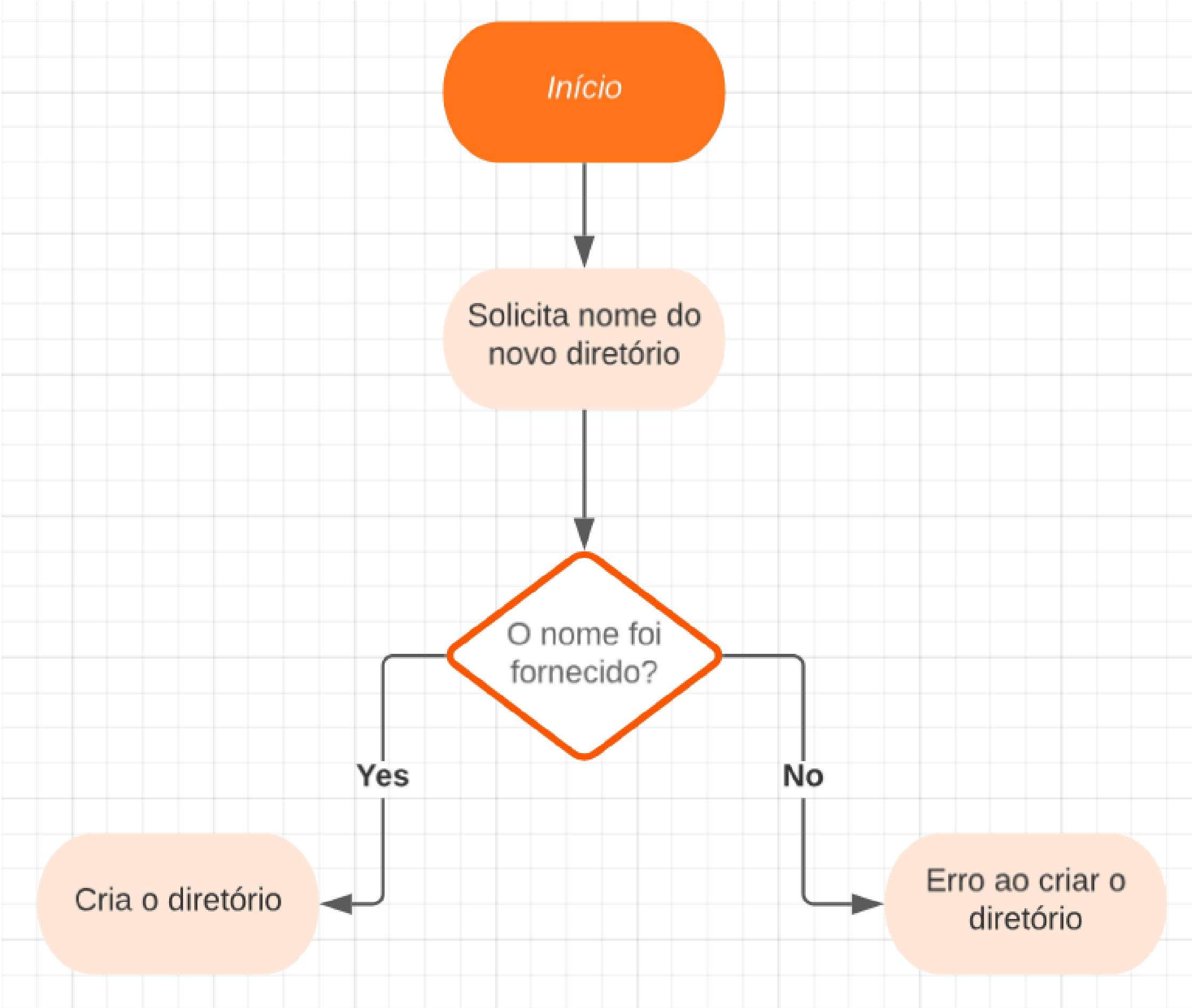
Função Criar diretório

Legenda do fluxorama

Contexto

Decisão

Funcionalidades



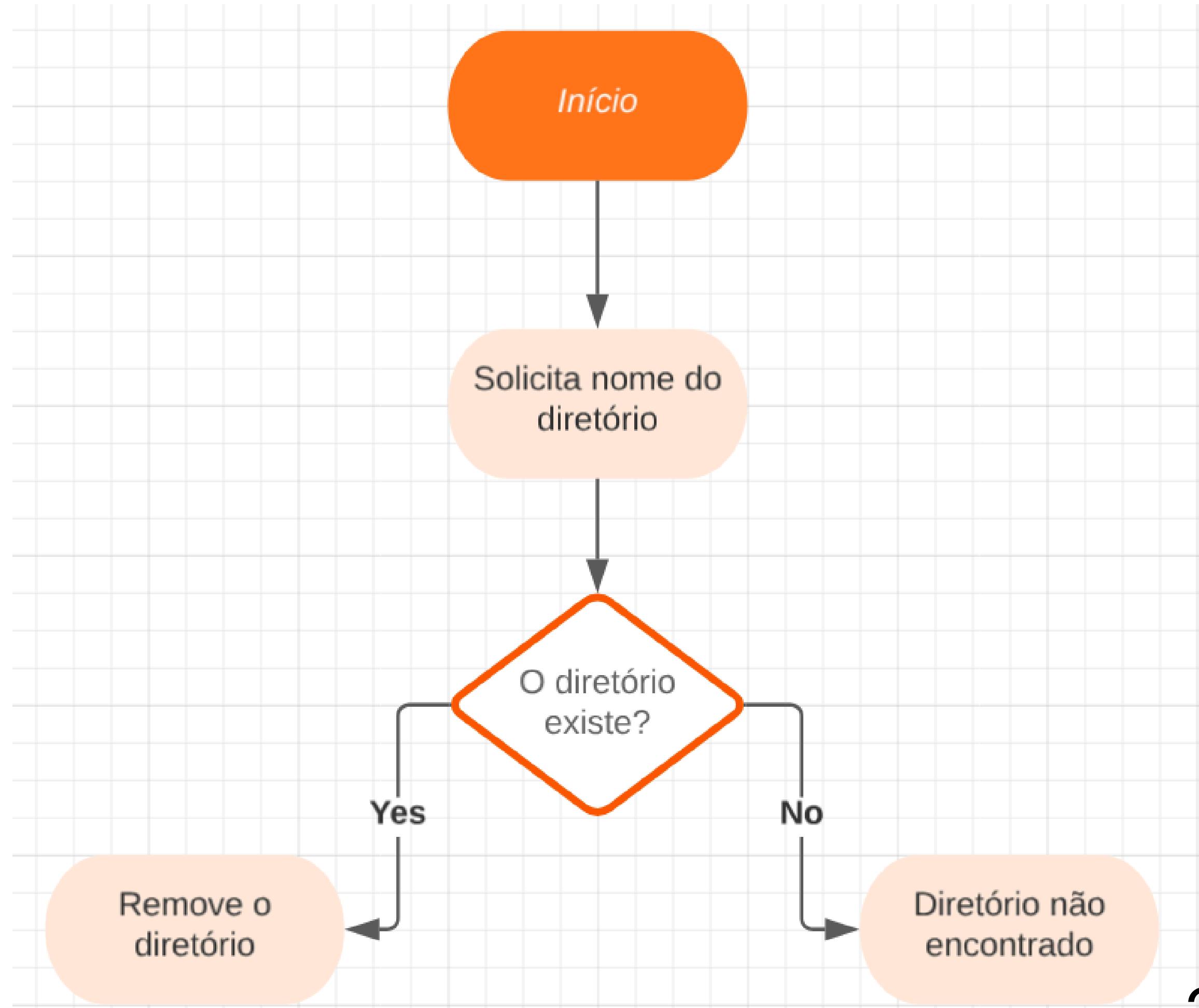
Função Remover diretório - I-Node

Legenda do fluxorama

Contexto

Decisão

Funcionalidades



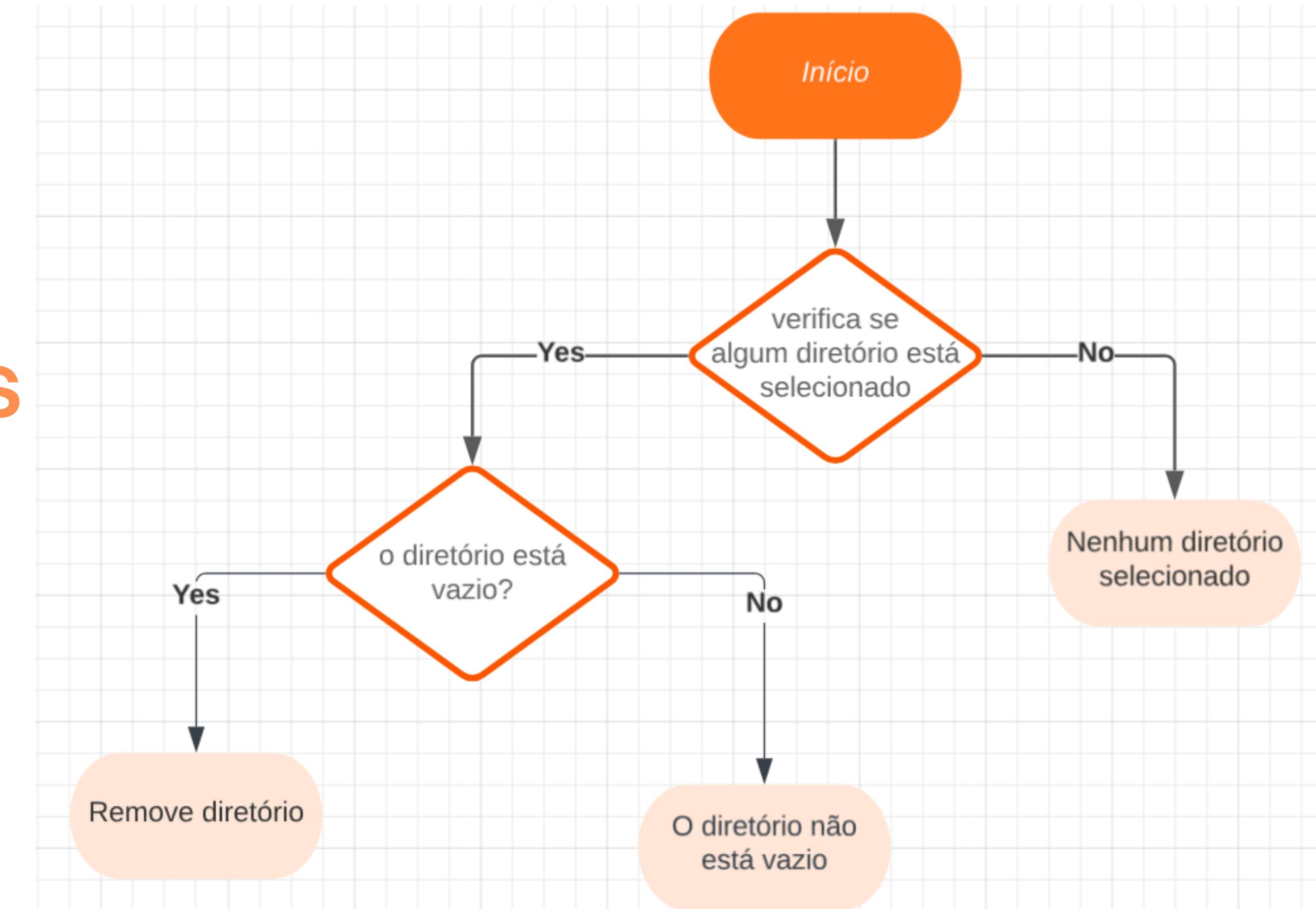
Função Remover diretório - tabelas de Blocos livres

Legenda do fluxograma

Contexto

Decisão

Funcionalidades



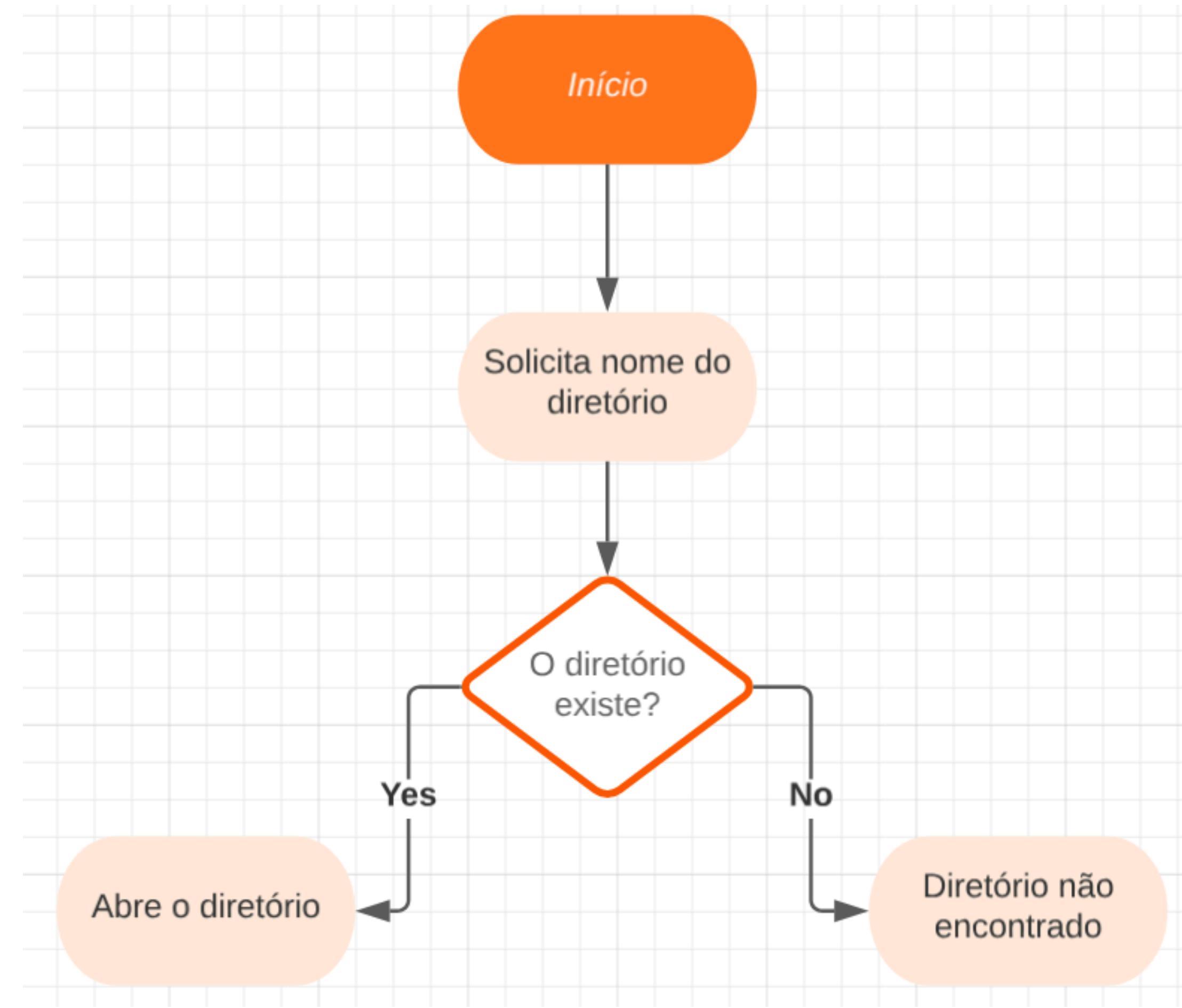
Função Abrir diretório - I-Node

Legenda do fluxorama

Contexto

Decisão

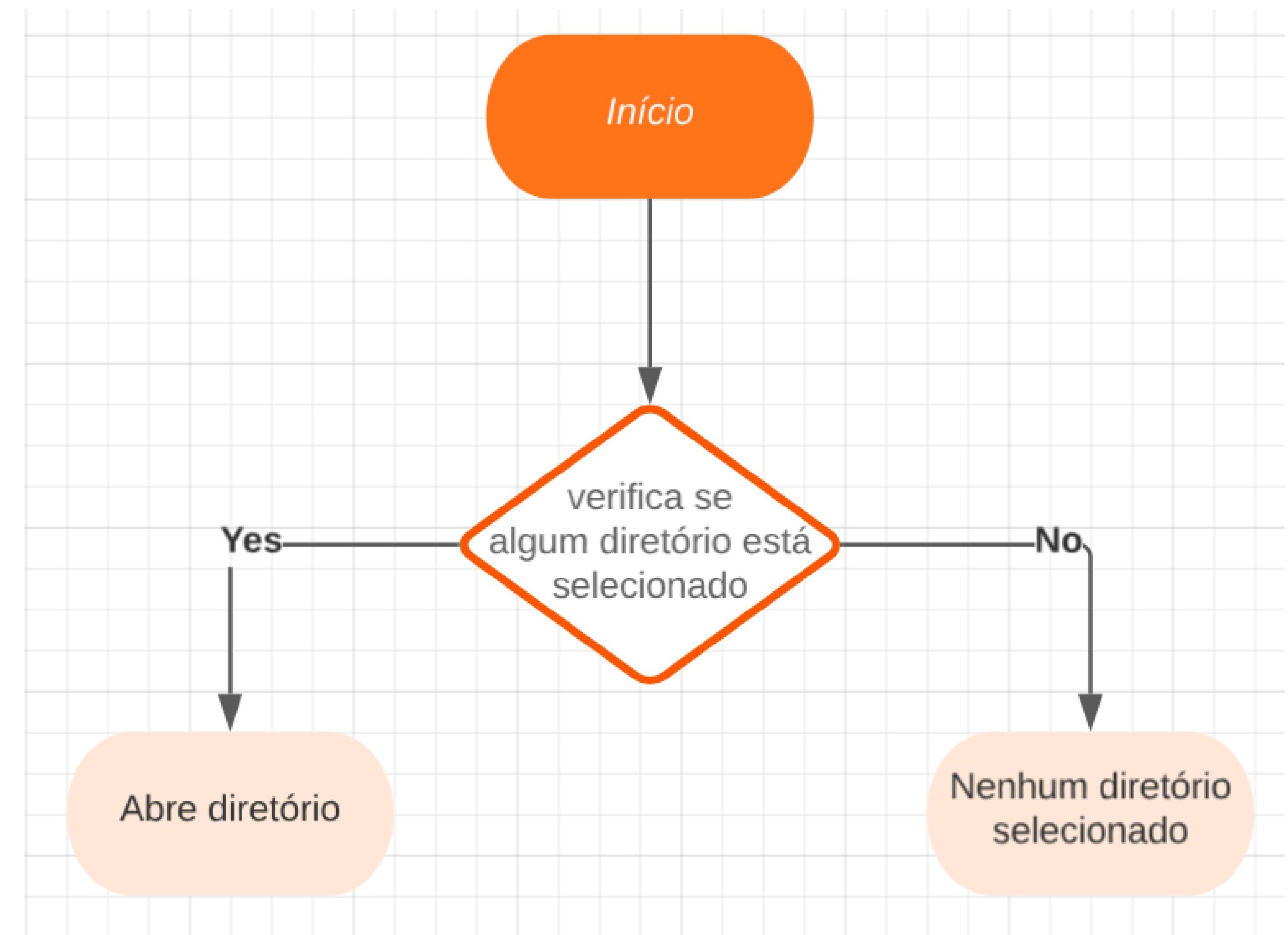
Funcionalidades



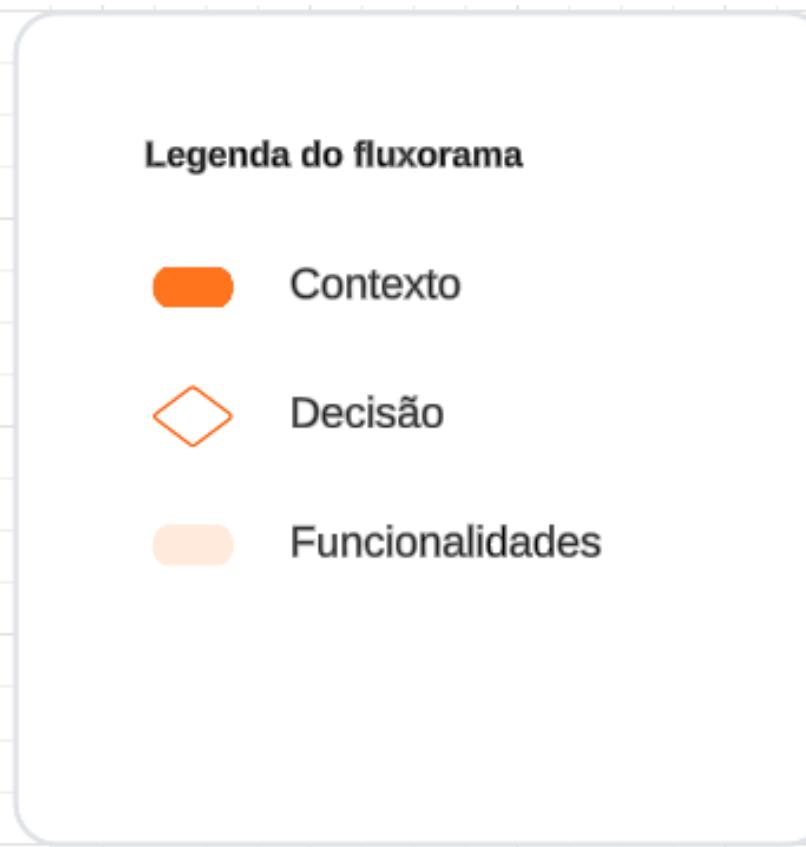
Função Abrir diretório - tabela de blocos livres

Legenda do fluxorama

- Contexto
- Decisão
- Funcionalidades



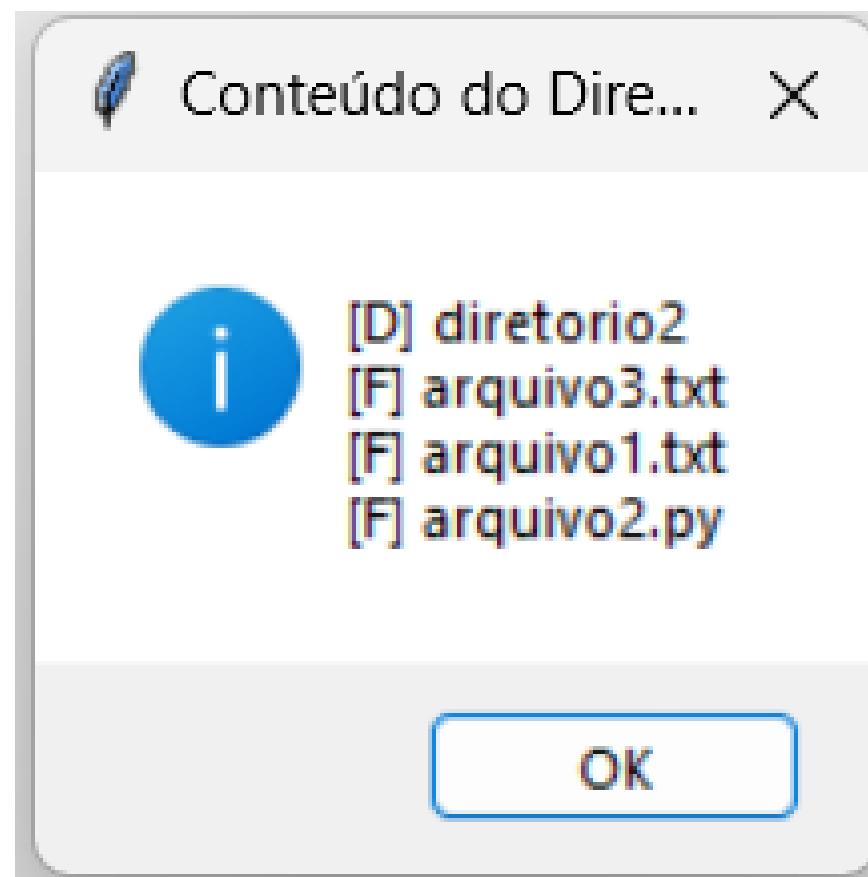
Função Diretório Anterior/Voltar



Função Listar Diretório

- 01 Lista o conteúdo do diretório atual.
- 02 Na aplicação em i-node, essa função retorna os objetos do diretório atual e a natureza deles;
- 03 Essa função não existe na aplicação em tabela de blocos livres pois o conteúdo é atualizado constantemente no visor pela função `Update_directory_listbox`.

Função Listar Diretório



Aplicação em I-node

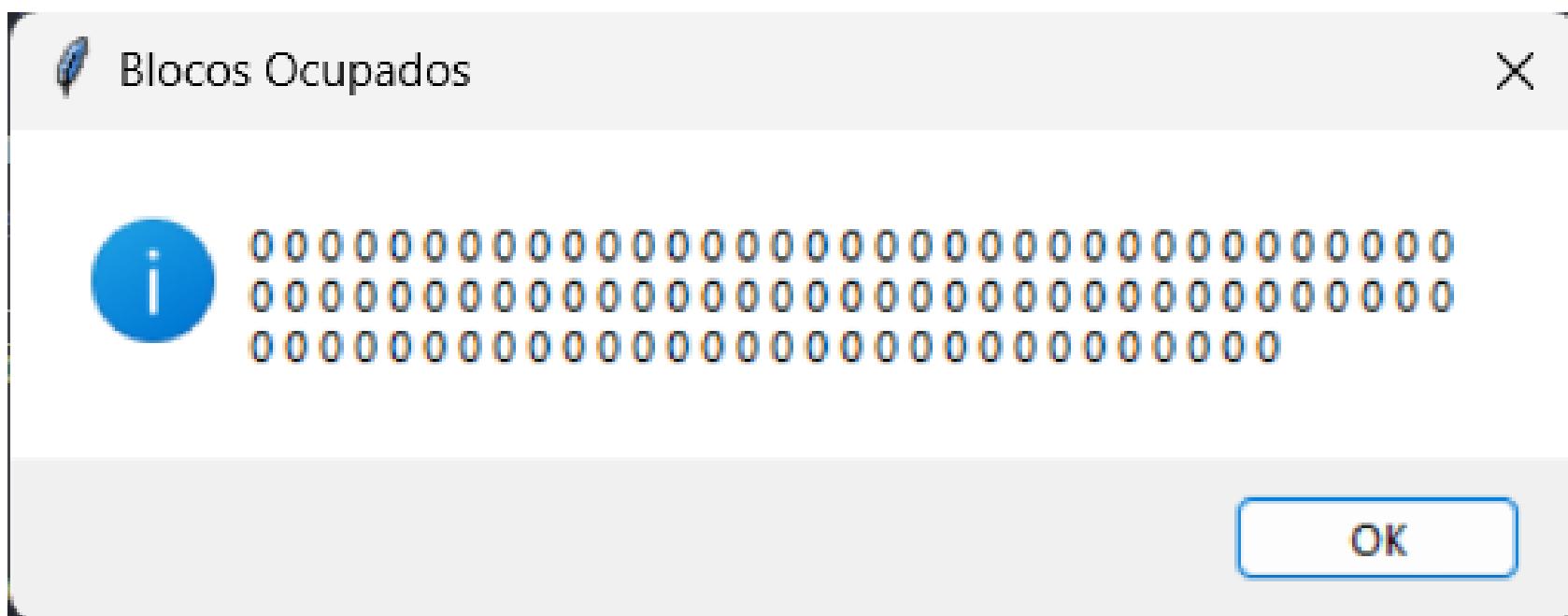
Diretório Atual: diretório1	
	arquivo1.txt
	diretório2
	arquivo3
	arquivo2.py

Tabela de blocos livres

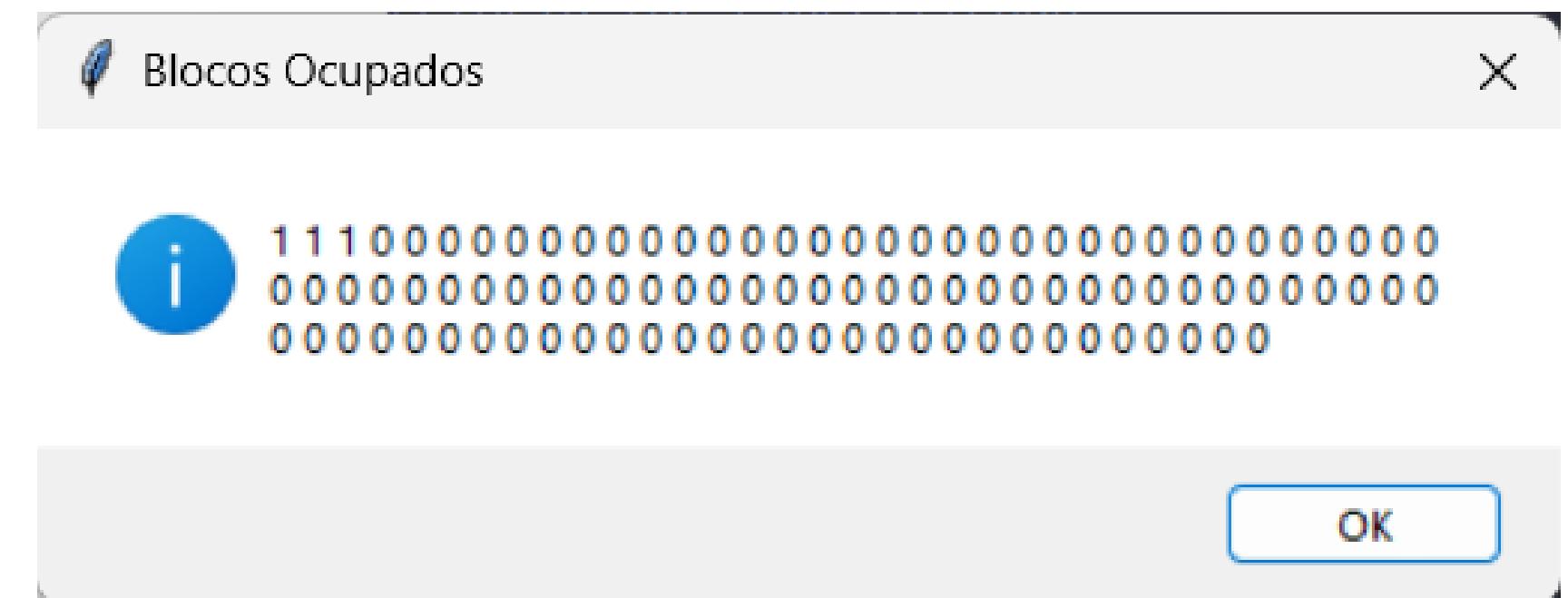
Função Visualizar Blocos Ocupados

- 01 **Essa função retorna uma representação em bitmap dos blocos alocados no sistema de arquivos.**
- 02 **Blocos alocados são representados por 1's e blocos livres são representados por 0's**

Função Visualizar Blocos Ocupados

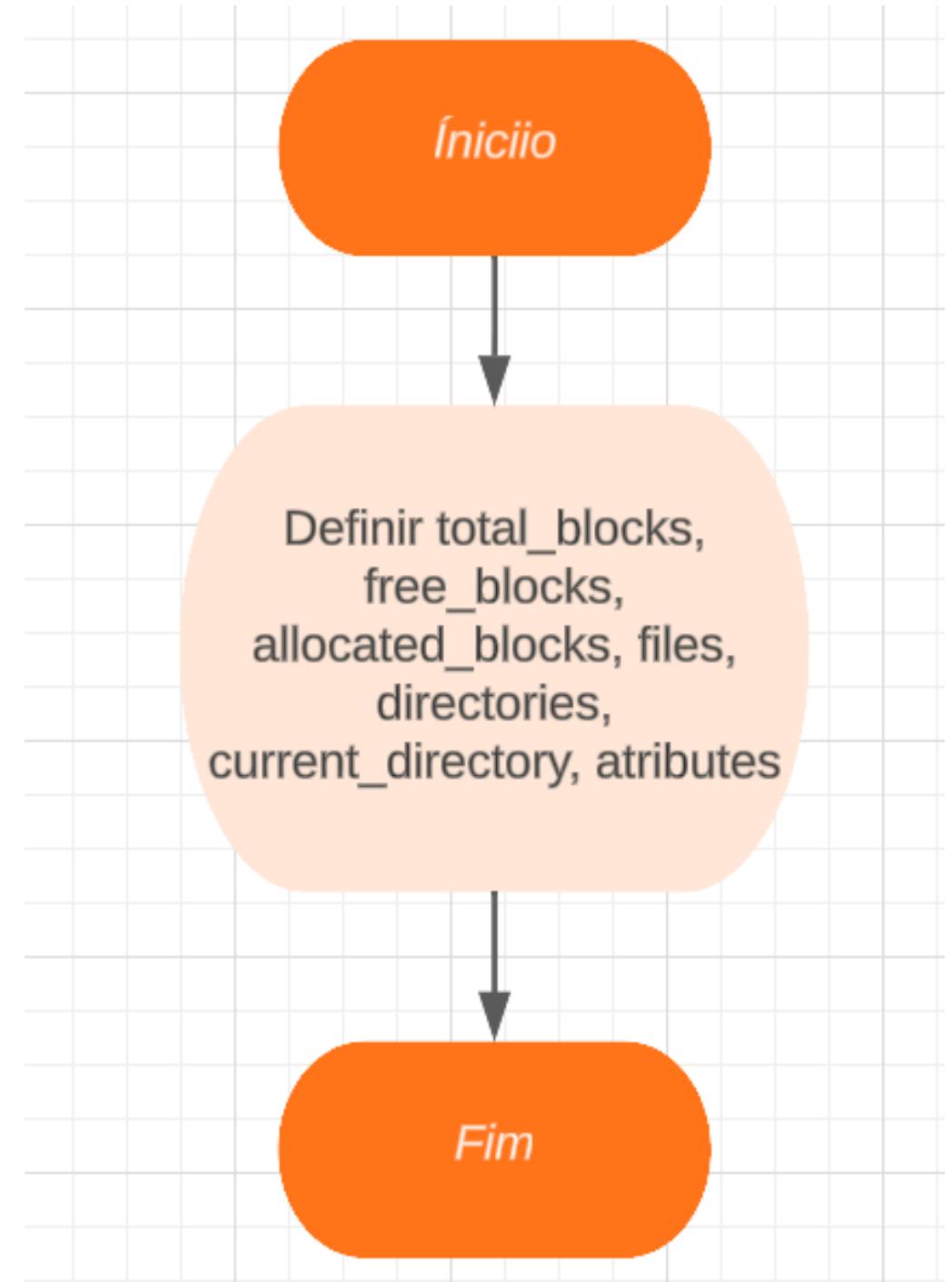


Todos os blocos estão livres.



Os blocos 0, 1 e 2 estão ocupados.

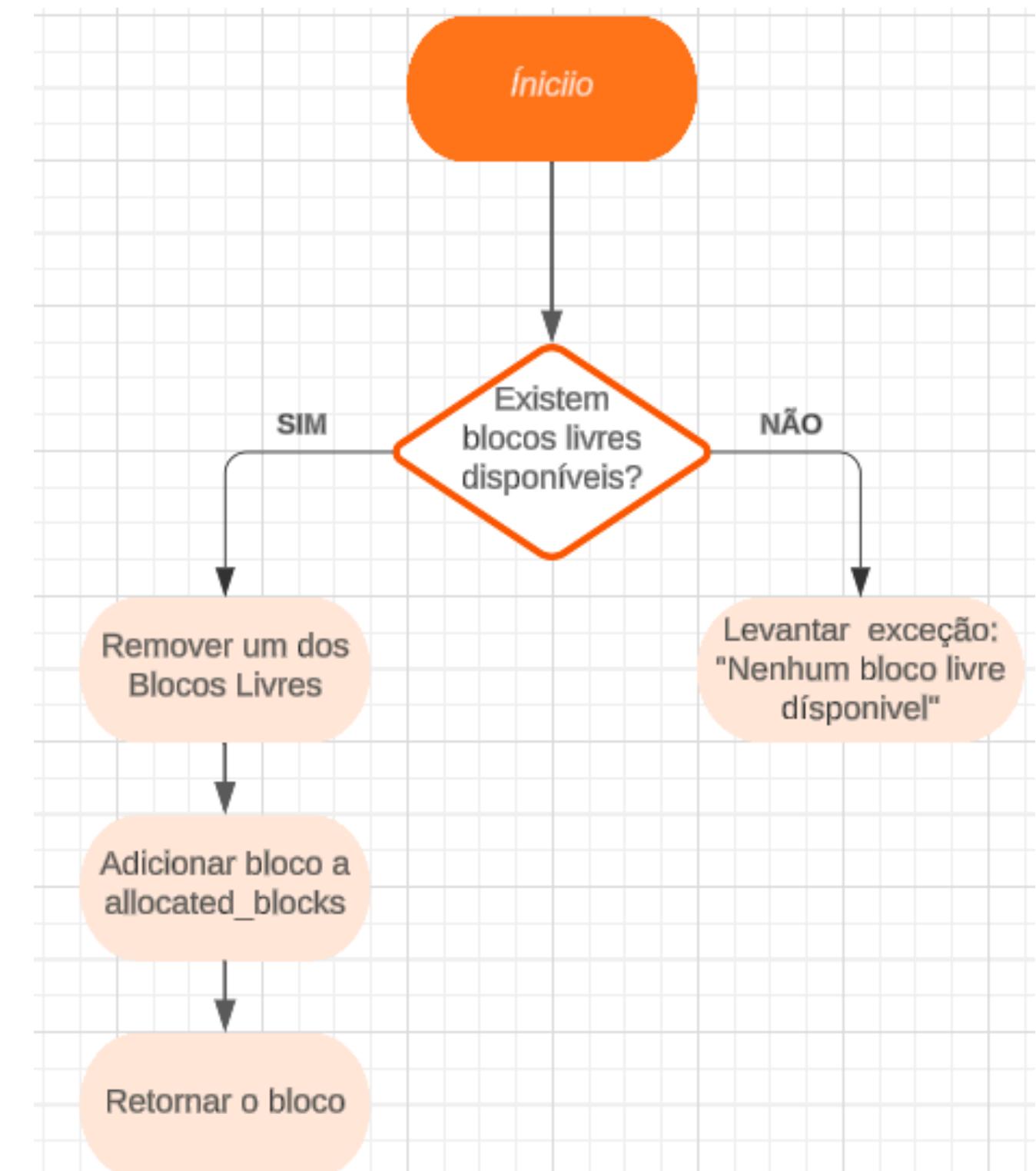
Método de Inicialização da Classe FileSystem (`__init__`)



Legenda do fluxorama

- Contexto
- Decisão
- Funcionalidades

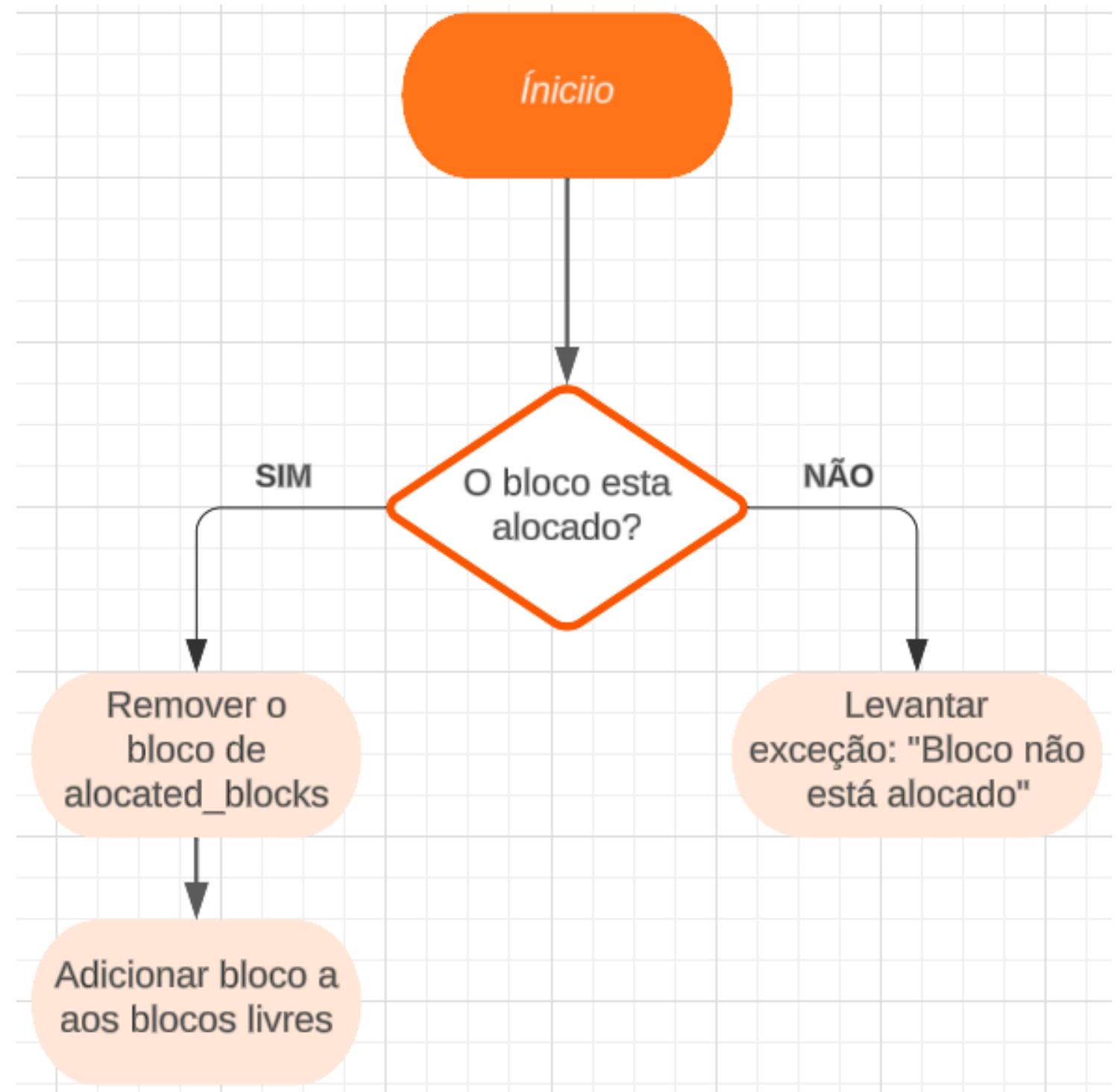
Função allocate_block (Alocar Bloco)



Legenda do fluxorama

- Contexto
- Decisão
- Funcionalidades

Função free_block (Liberar Bloco)



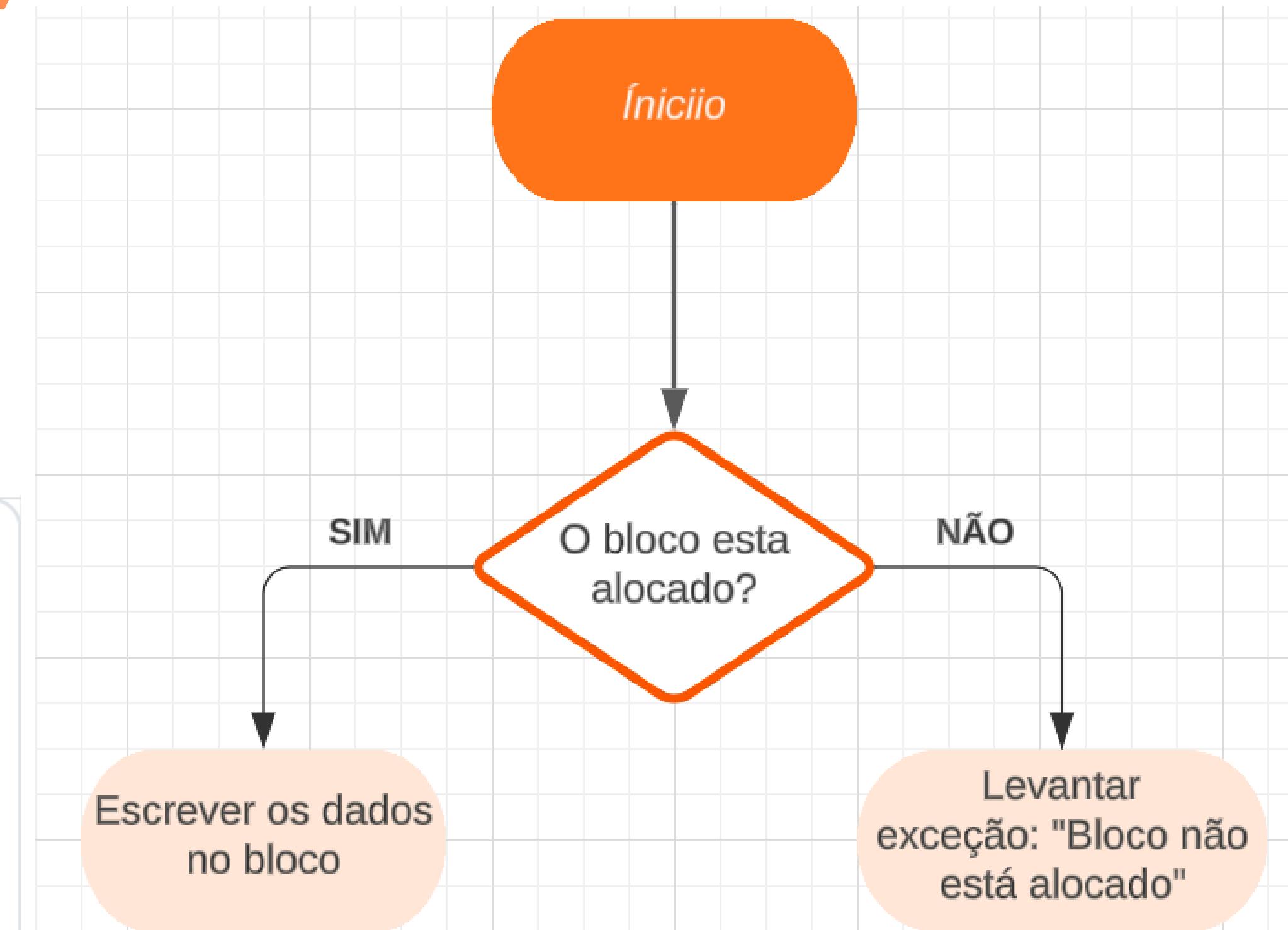
Legenda do fluxograma

- Contexto
- Decisão
- Funcionalidades

Função write_block (Escrever no Bloco)

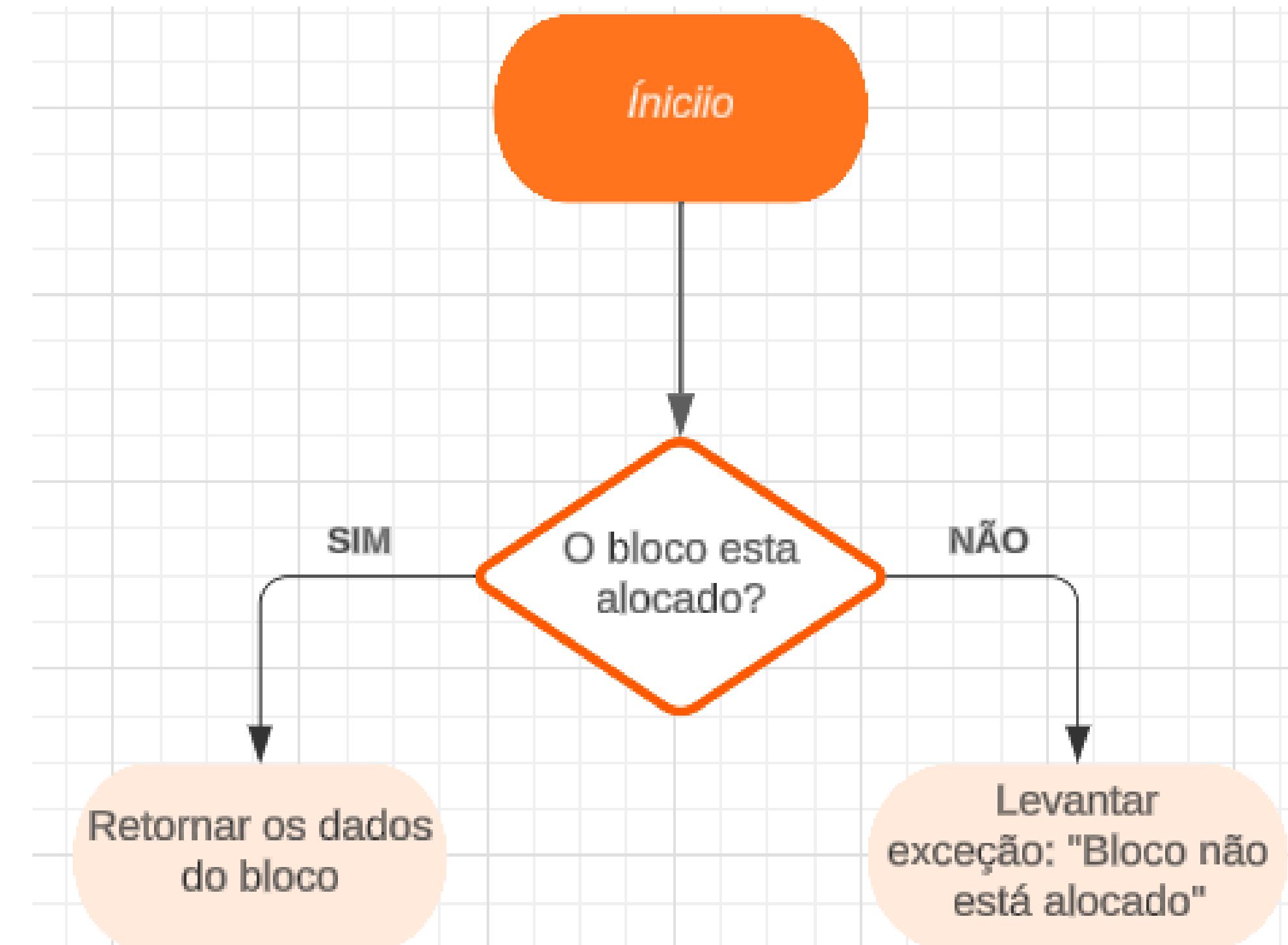
Legenda do fluxograma

- Contexto
- Decisão
- Funcionalidades

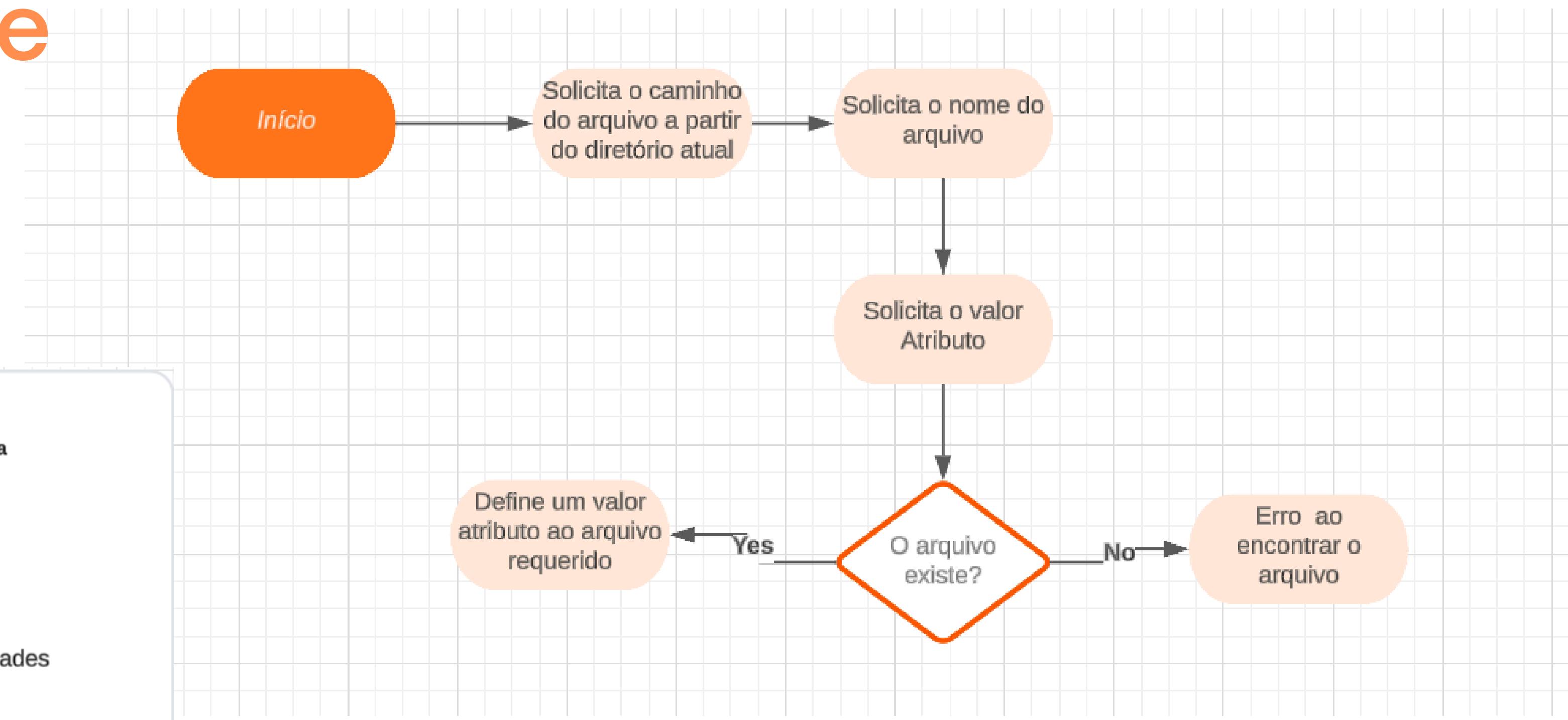


Função read_block (Ler do Bloco)

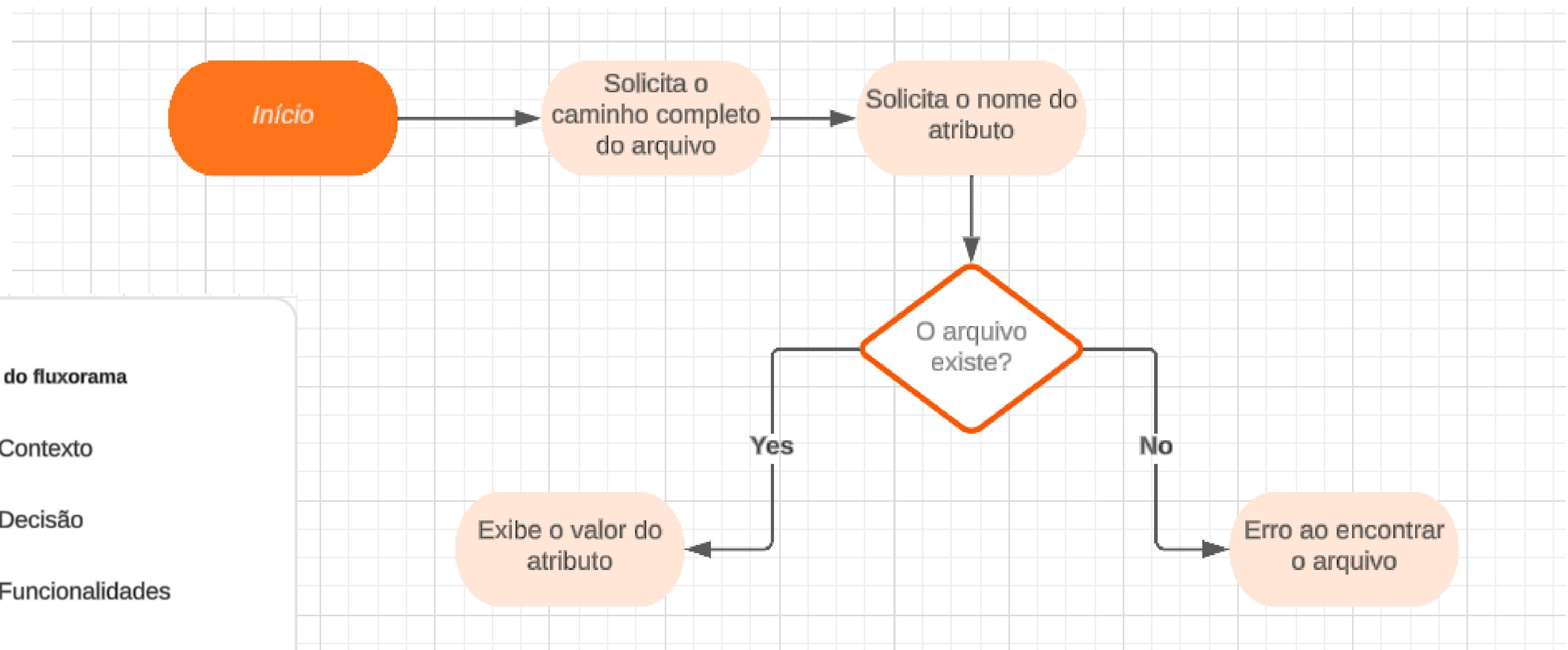
- Legenda do fluxograma
- Contexto
 - Decisão
 - Funcionalidades



Função Set attribute-I-Node



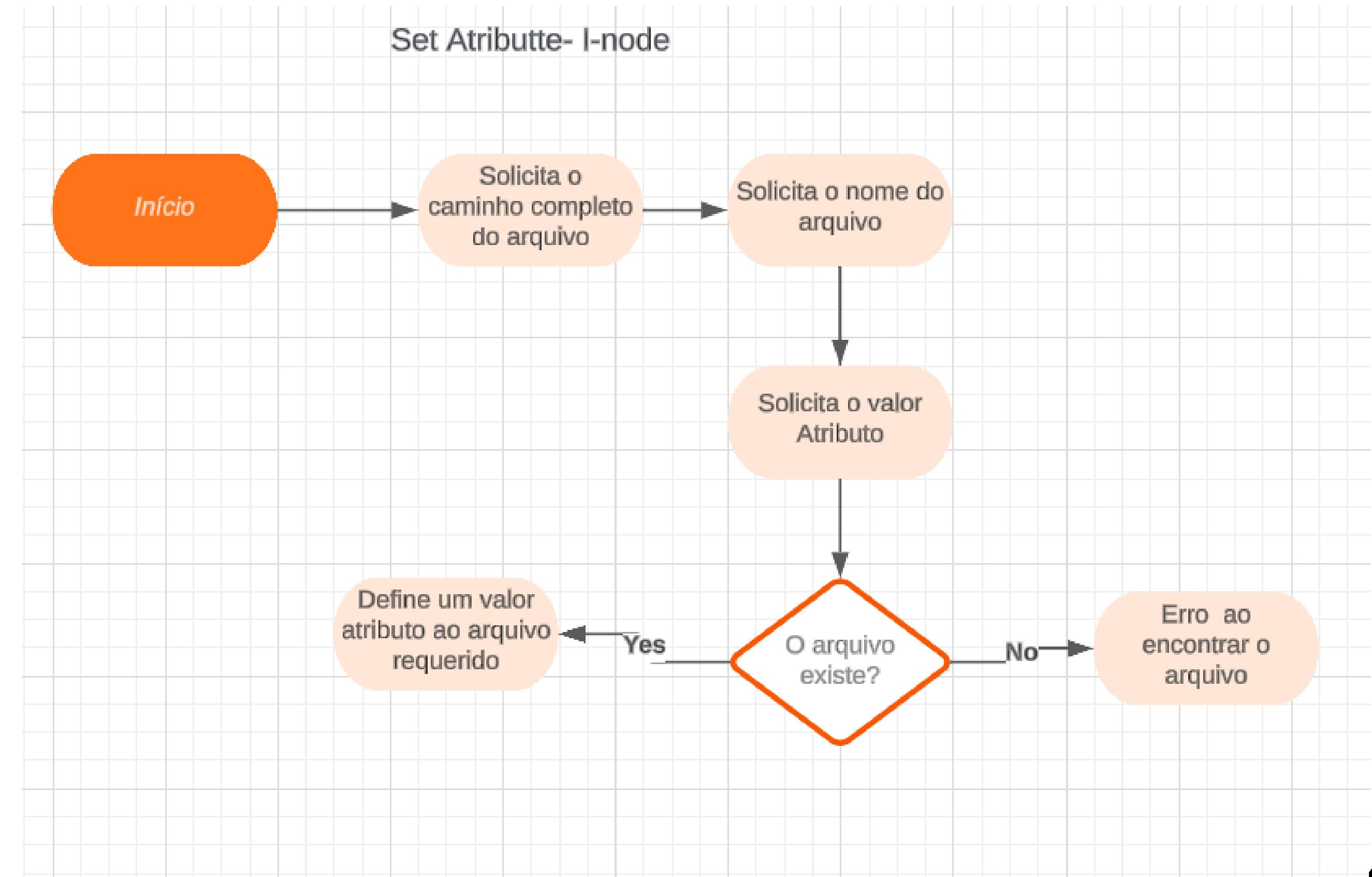
Função Get attribute-I-Node



Função Get attribute- I-Node

Legenda do fluxograma

- Contexto
- Decisão
- Funcionalidades



Reconhecimentos e Direitos Autorais

@autor: [Leônidas Serra, Isabela Oliveira, Paulo Arthur, Samuel Basílio]

@ contato: [leonidas.serra@discente.ufma.br, moreira.isabela@discente.ufma.br, paulo.alb@discente.ufma.br]

@data última versão: [14/06/2024]

@versão: 1.0

@outros repositórios: [URLs - apontem para o seus gits, se quiserem]

@Agradecimentos: Universidade Federal do Maranhão (UFMA), Professor Doutor Thales Levi Azevedo Valente, e colegas de curso.

@Copyright/License

Este material é resultado de um trabalho acadêmico para a disciplina SISTEMAS OPERACIONAIS, sobre a orientação do professor Dr. THALES LEVI AZEVEDO VALENTE, semestre letivo 2024.1, curso Engenharia da Computação, na Universidade Federal do Maranhão (UFMA). Todo o material sob esta licença é software livre: pode ser usado para fins acadêmicos e comerciais sem nenhum custo. Não há papelada, nem royalties, nem restrições de "copyleft" do tipo GNU. Ele é licenciado sob os termos da licença MIT reproduzida abaixo e, portanto, é compatível com GPL e também se qualifica como software de código aberto. É de domínio público. Os detalhes legais estão abaixo. O espírito desta licença é que você é livre para usar este material para qualquer finalidade, sem nenhum custo. O único requisito é que, se você usá-lo, nos dê crédito.

Copyright © 2024 Educational Material

Este material está licenciado sob a Licença MIT. É permitido o uso, cópia, modificação, e distribuição deste material para qualquer fim, desde que acompanhado deste aviso de direitos autorais.

O MATERIAL É FORNECIDO "COMO ESTÁ", SEM GARANTIA DE QUALQUER TIPO, EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO ÀS GARANTIAS DE COMERCIALIZAÇÃO, ADEQUAÇÃO A UM DETERMINADO FIM E NÃO VIOLAÇÃO. EM HIPÓTESE ALGUMA OS AUTORES OU DETENTORES DE DIREITOS AUTORAIS SERÃO RESPONSÁVEIS POR QUALQUER RECLAMAÇÃO, DANOS OU OUTRA RESPONSABILIDADE, SEJA EM UMA AÇÃO DE CONTRATO, ATO ILÍCITO OU DE OUTRA FORMA, DECORRENTE DE, OU EM CONEXÃO COM O MATERIAL OU O USO OU OUTRAS NEGOCIAÇÕES NO MATERIAL.