

# UNIVERSIDADE SÃO TOMÁS DE MOÇAMBIQUE

# FACULDADE DE CIÊNCIAS E DE TECNOLOGIAS DE INFORMAÇÃO

## LICENCIATURA EM DESENVOLVIMENTO DE SOFTWARE

### PROGRAMAÇÃO I

Turma: 1P2LDS1

**Relatório sobre**

**Trabalho Prático I**

**Plataforma de Gestão de Propriedades**

*Edilson Alexandre Cuamba*

**Orientador:** Eng. Carlos Victorino Padeiro

Maputo, Dezembro de 2021

# **Informações de Contacto**

**Nome:** Edilson Alexandre Cuamba

**Código:** 2021 1010 69

**Telefone:** (+258) 84 24 73 772 & (+258) 82 25 65 148

**Email:** [edilsoncuamba@gmail.com](mailto:edilsoncuamba@gmail.com)

Faculdade De Ciências E Tecnologias De Informação da Universidade São Tomás de Moçambique

índice

[**UNIVERSIDADE SÃO TOMÁS DE MOÇAMBIQUE** 1](#_Toc92032022)

[**FACULDADE DE CIÊNCIAS E DE TECNOLOGIAS DE INFORMAÇÃO** 1](#_Toc92032023)

[**LICENCIATURA EM DESENVOLVIMENTO DE SOFTWARE** 1](#_Toc92032024)

[PROGRAMAÇÃO I 1](#_Toc92032025)

[**Informações de Contacto** 2](#_Toc92032026)

[Documentação de Software 4](#_Toc92032027)

[Introdução 5](#_Toc92032028)

[Objetivos Gerais 5](#_Toc92032029)

[Objetivos Específicos 5](#_Toc92032030)

[Desenvolvimento 6](#_Toc92032031)

[O que foi utilizado para desenvolver o software? 6](#_Toc92032032)

[Que lógica foi usada para pensar no problema? 6](#_Toc92032033)

[Começamos a entender os requisitos. 6](#_Toc92032034)

[Diagramas UML 6](#_Toc92032035)

[Casos de Uso 6](#_Toc92032036)

[Classes 7](#_Toc92032037)

[Actividades 7](#_Toc92032038)

[Fluxogramas 8](#_Toc92032039)

[calcular a quantidade de vivendas e flats vendidas, a quantidade total vendida; 8](#_Toc92032040)

[Determinar qual local facturou mais, acompanhado do valor; 10](#_Toc92032041)

[Calcular o valor total recebido pela empresa (tendo em conta os descontos). 11](#_Toc92032042)

[Calcular o lucro da empresa, sabendo que 70% do valor total é utilizado em despesas (material, trabalhadores e outros); 12](#_Toc92032043)

[Calcular o valor total dos descontos, sabendo que em todas flats do tipo 3 existe um desconto de 10% e dotipo 2 existe um desconto de 5% sobre o valor da propriedade. 14](#_Toc92032044)

[visualizar os dados lidos e calculados na forma de tabela com cabeçalho de seguinte maneira (exemplo): 15](#_Toc92032045)

[Calcular o valor total recebido em vivendas e flats. 16](#_Toc92032046)

[Conclusão 19](#_Toc92032047)

# 

# Documentação de Software

Antes de apresentar a documentação do software, “Plataforma de Gestão de Propriedades”, gostaríamos de explicar o que é e qual é a importância da documentação de software.

A documentação de software é um texto explicativo expositivo que é feito baseado no software desenvolvido, este documento que por sua vez irá acompanhar o software por toda a sua existência podendo ser alterado ou editado, apresenta varias informações importantes sobre o software, por exemplo ele explica como utilizar, como o software foi feito, e outros detalhes essências para qualquer pessoas que seja ligada a programação senha uma noção de com o software foi desenvolvido.

Neste breve relatório, faremos uma breve documentação do software em questão que foi desenvolvido.

# Introdução

Antes mesmo de mostrar como o software foi contruído gostaría de mostrar que tecnologias usaram e quais os objetivos do trabalho no geral e no especifico.

## Objetivos Gerais

Expor de forma clara e concisa como o software foi desenvolvido e como ele atende as expectativas do cliente.

## Objetivos Específicos

* De forma clara e direta falar das tecnologias usadas e de como o software funciona;
* Explicar os métodos principais do software;
* O motivo de criação de certas variáveis e métodos;
* E os porquês de certos laços usados.

# Desenvolvimento

## O que foi utilizado para desenvolver o software?

Para desenvolver está plataforma foi usado a linguagem **Java**, esta linguagem foi desenvolvida por James Gosling na década de 90 como propriedade da Sun Microsystems, em 2005 a Sun Microsystems foi comprada pela Oracle e hoje a linguagem Java é propriedade dela.

Foi usado também algumas ferramentas do mundo Java como o gestor de projecto Apache Maven e algumas dependências como a JColor para impressão de textos coloridos no console e Ascii-table para criação de tabelas para impressão no console.

## Que lógica foi usada para pensar no problema?

Para resolver o problema do cliente decidimos fazer uma analise profundo do problema e pensar nos melhores meios de desenvolver um software e ainda compartilhar o código. Para conseguirmos esse feito utilizei a UML para criar diagramas e pensar em como resolveria o problema.

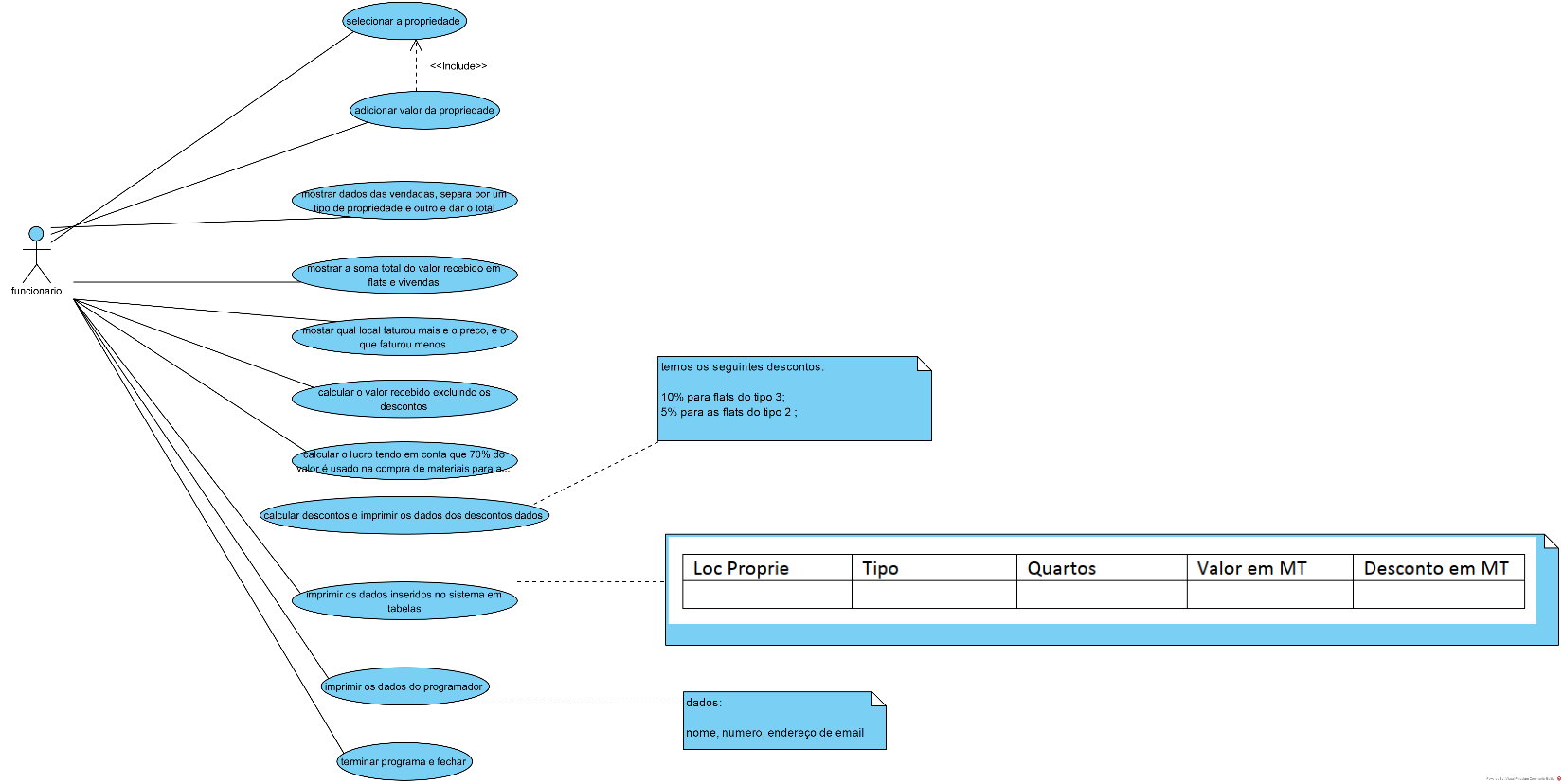
## Começamos a entender os requisitos.

Para este projeto o cliente queria que construíssemos uma aplicação que pudesse gerir as propriedades que estão sendo vendidas e com esses dados ele queria pegar algumas informações, alguns relatórios.

### Diagramas UML

#### Casos de Uso

Para entender bem o problema fizemos um diagrama de casos de uso que mostra a interação do utilizador da plataforma com a plataforma, ele apresenta quais as funcionalidades ele terá acesso que quais funcionalidades são dependentes umas das outras.



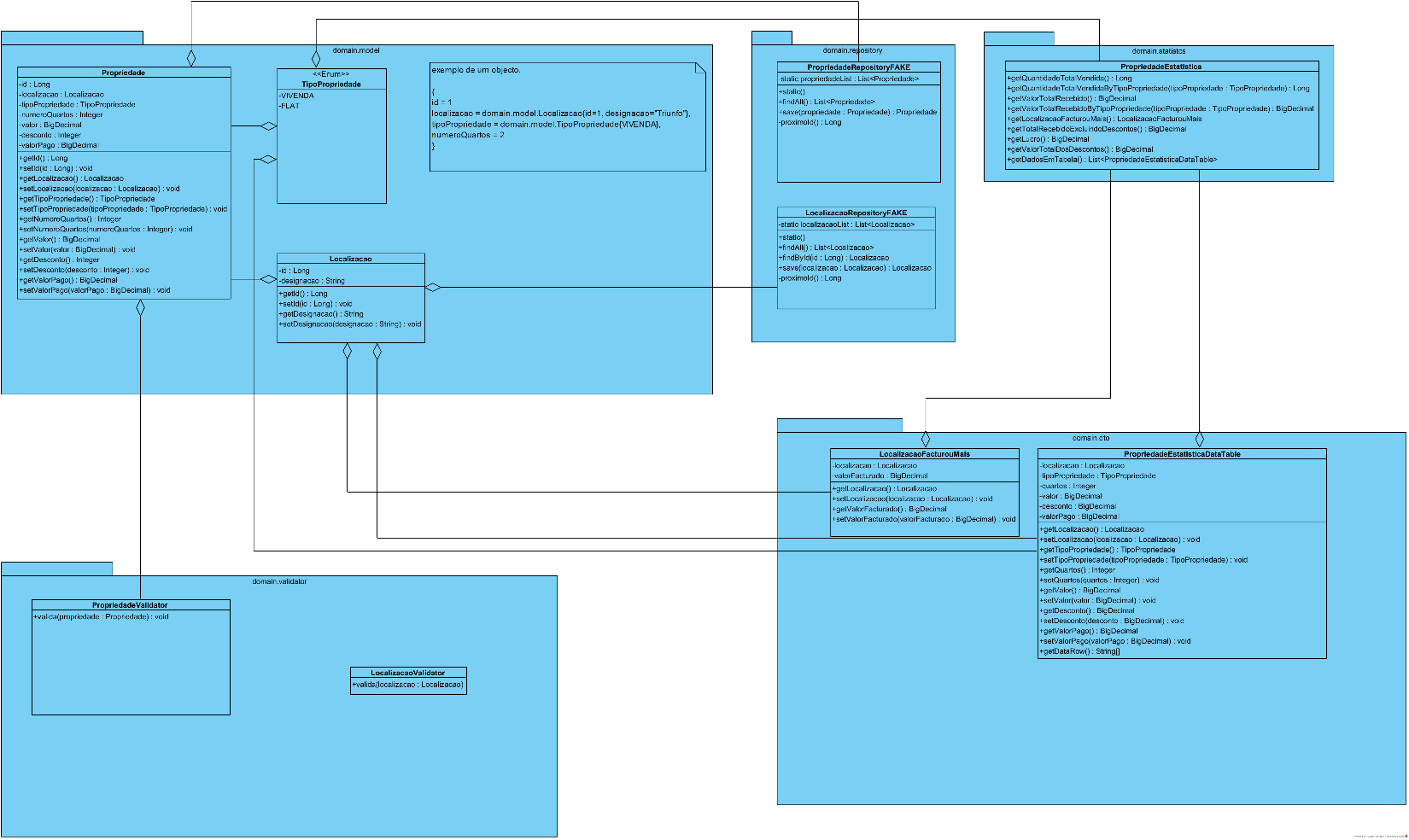
#### Classes

Depois de entender as funcionalidades procuramos entender como as informações estariam organizadas em um sistema feito usando uma linguangem orientada a objecto.

Percebemos que temos vários objectos no nosso domínio de problema, sendo o mais importante o objecto propriedade.

A propriedade é o motivo pelo qual estamos fazendo está plataforma. Decidimos que a Propriedade seria uma classe que se relacionaria com outras classes menos importantes sozinhas como por exemplo a classe Localização que representa uma localização de uma propriedade, também percebemos que o cliente quer guardar com as propriedades os seus tipos como Flat e Vivendo, como estes valores são fixos e imutáveis, decidimos usar enumerations. E por fim usamos tipos de dados normais para armazenar dados como valores monetários e descontos.

Podemos ver na imagem.



Aqui temos os diagramas para descrever como funcionaria as classes que interagiriam directamente com o utilizador:~

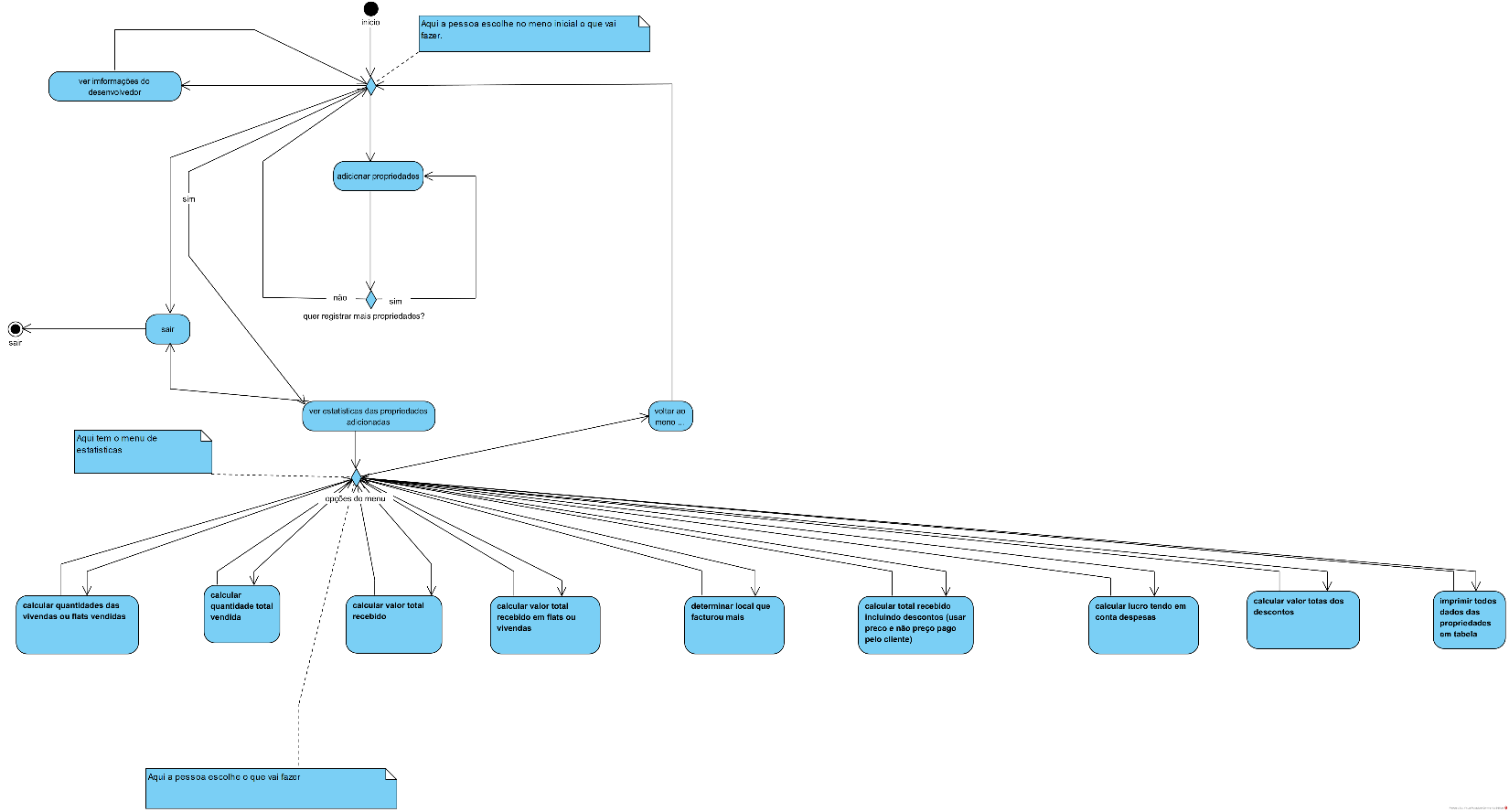


#### Actividades

No diagrama de actividades, é onde descrevemos como e a ordem em que as actividades serão desenvolvidas ao decorrer do tempo.

Neste diagrama colocamos em ordem como as actividade devem ser feitas no sistema.

Decidimos que o utilizador ao aceder ao sistema terá acesso á um menu principal com algumas funcionalidades distintas e que teria um menu mais profundo com as funcionalidades de extração de dados.



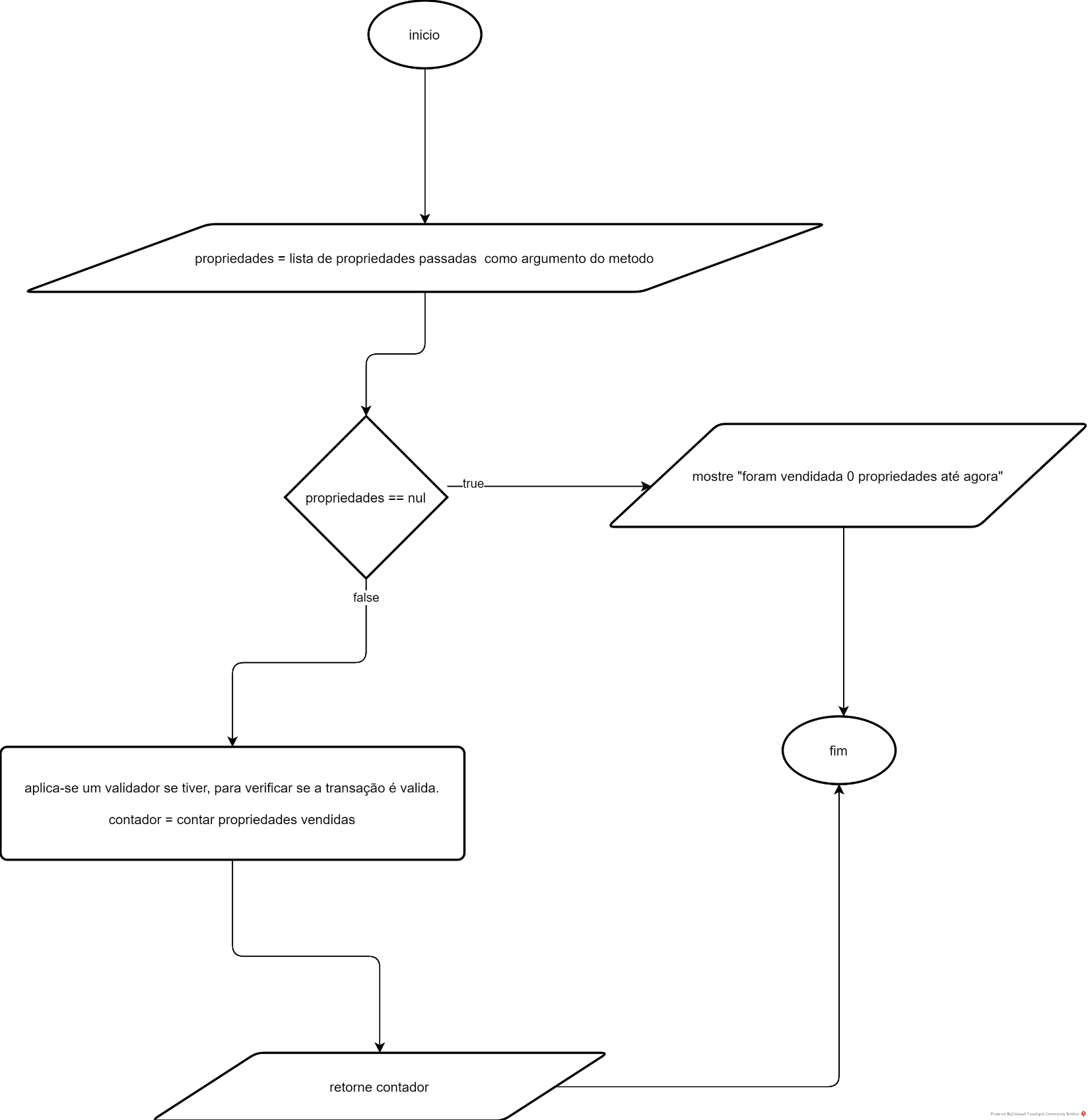
#### Fluxogramas

Os fluxogramas estão dividido em funcionalidades que o cliente pediu para as estatísticas, como por exemplo “Calcular a quantidade de propriedades vendidas”, aqui teremos todas as funcionalidades.

##### calcular a quantidade de vivendas e flats vendidas, a quantidade total vendida;

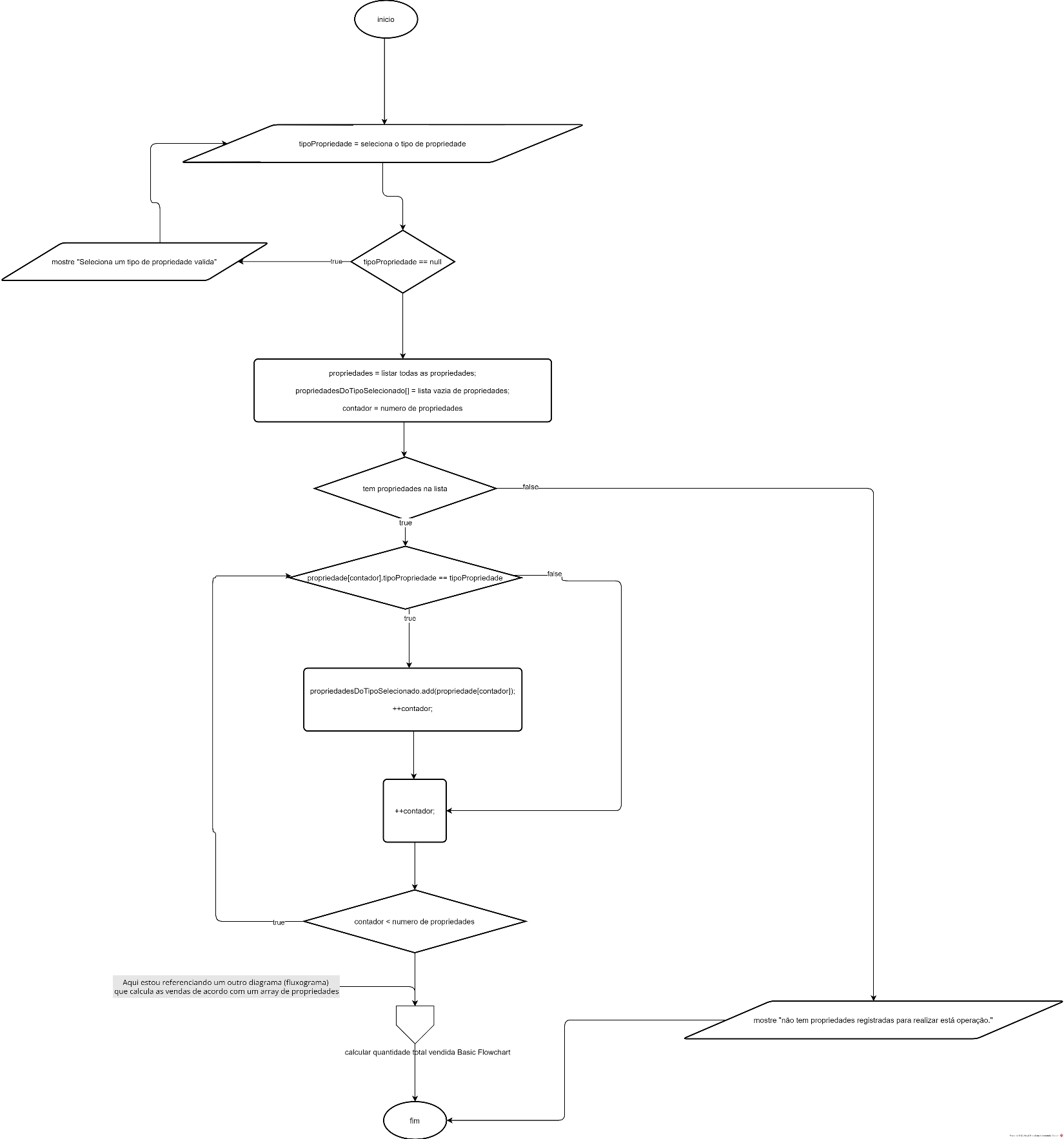
Para resolver este problema decidimos usar uma abordagem de reutilização de código, queríamos fazer métodos que fossem reutilizáveis por isso os dados são passados como array no parâmetro do método e esse array é processado.

Criamos um método simples que faz somente para qualquer lista de dados:



Apos isso derificamos se o array tem dados e por fim contamos as propriedade e retornamos o valor.

Agora a implementação que usamos para selecionar caso a pessoa queira de um tipo de propriedade especifica.

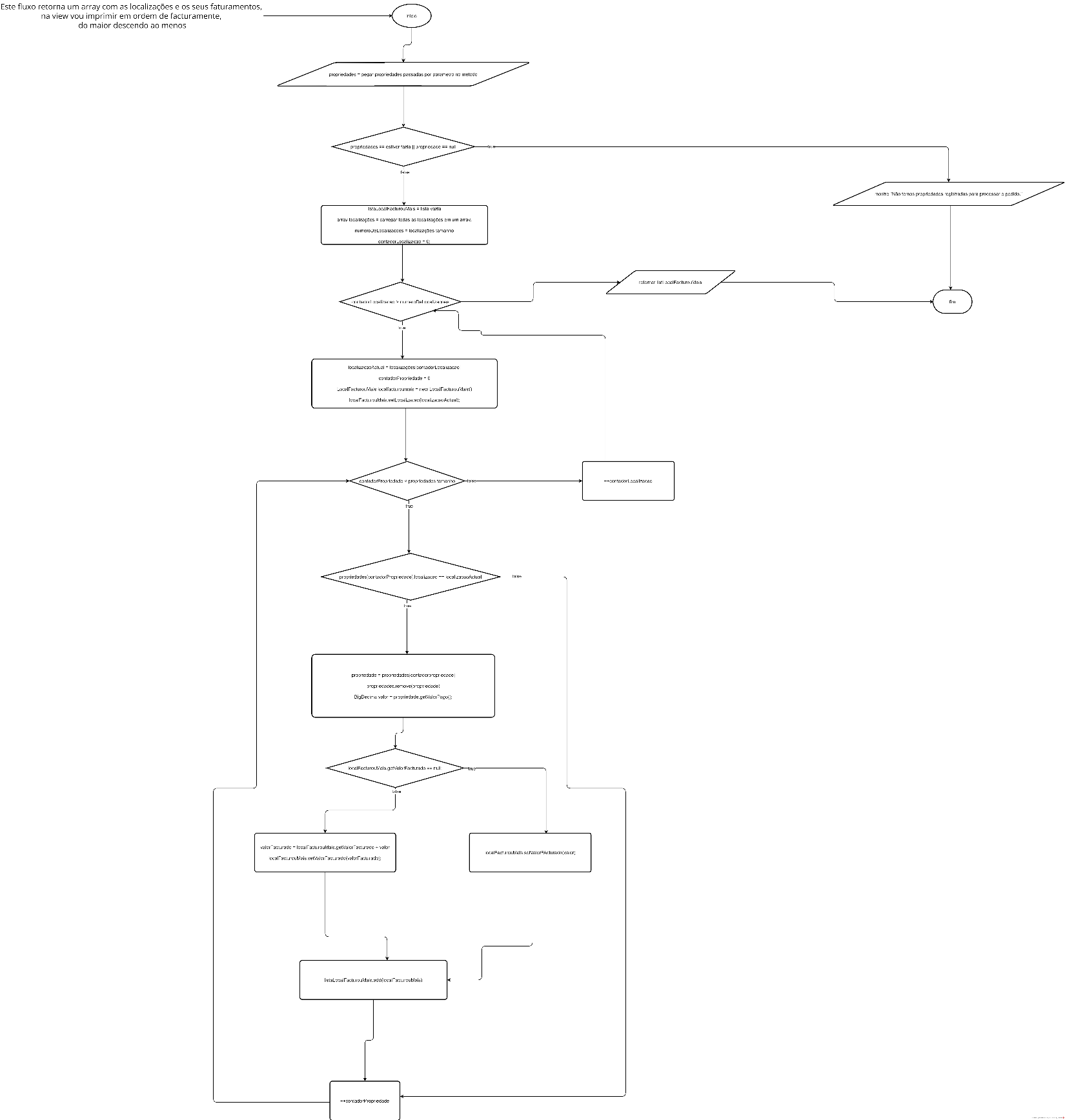


Então temos uma situação assim, um método recebe um array de dados e trabalha sobre ele e temos um outro método que recebe também uma lista e permite que o utilizador seleciona um tipo de propriedade, com esse tipo de propriedade em mãos o método faz uma procura e obtem todas as propriedades desse tipo e passa ao método principal.

##### Determinar qual local facturou mais, acompanhado do valor;

Para determinar o local que facturou mais optamos por fazer um fluxo que verifica o facturamento de cada localização. Pegavamos as localizações e depois pegamos uma e procuramos todas as propriedades dessa localização, apos isso somamos os valores ganhos e por fim comparamos os valores de todas as localizações.

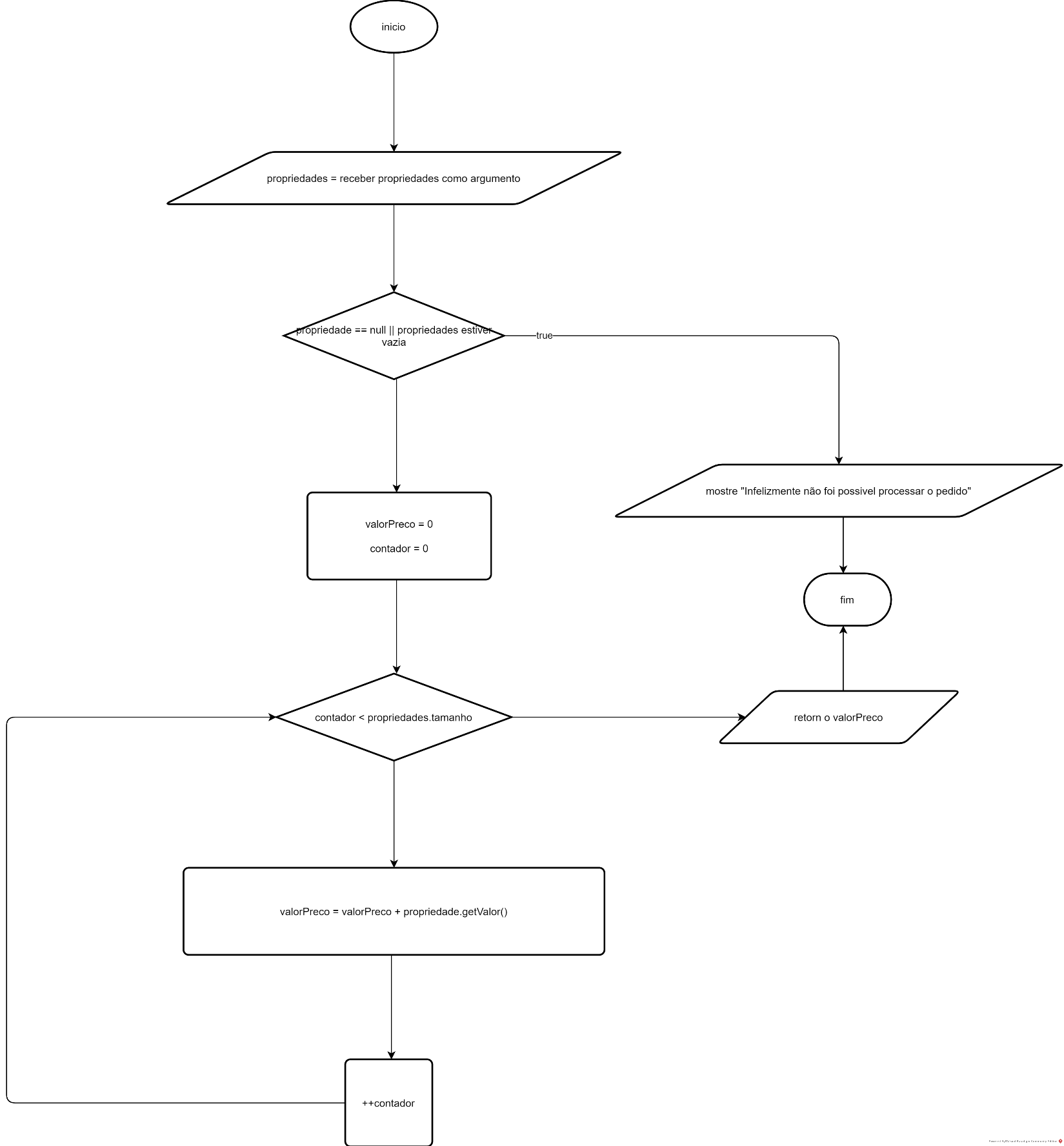
Veja no diagrama como ficou:



##### Calcular o valor total recebido pela empresa (tendo em conta os descontos).

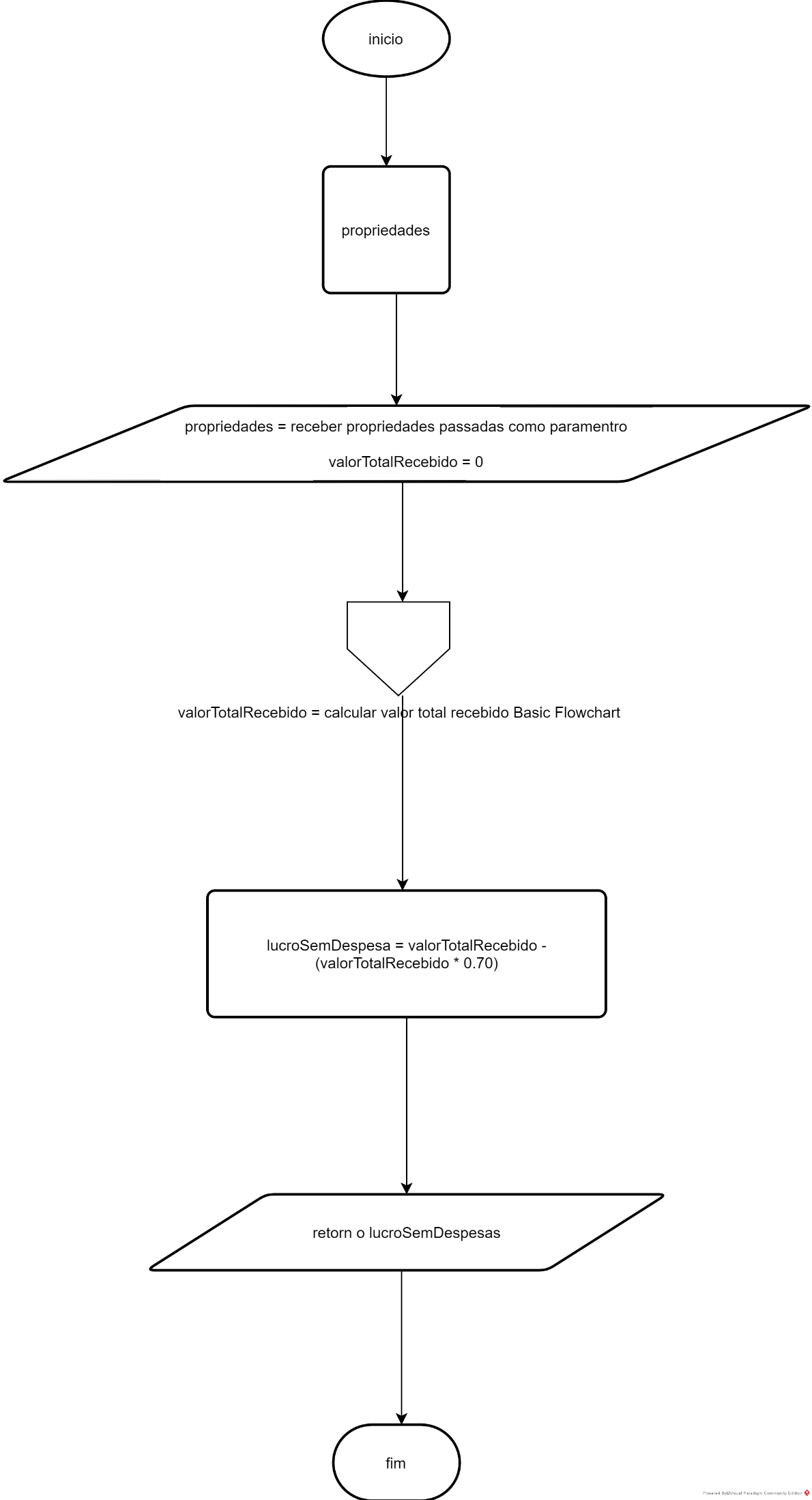
Neste caso devíamos criar uma funcionalidade que imprimisse o valor total recebido incluindo os descontos, isso significa que seria o valor da propriedade em si sem tirar os descontos e não o valor que o cliente pagou literalmente.

Para fazer essa funcionalidade decidimos criar um método que vai receber as propriedades somar os valor preços da propriedades e retornar esse valor, simples.



##### Calcular o lucro da empresa, sabendo que 70% do valor total é utilizado em despesas (material, trabalhadores e outros);

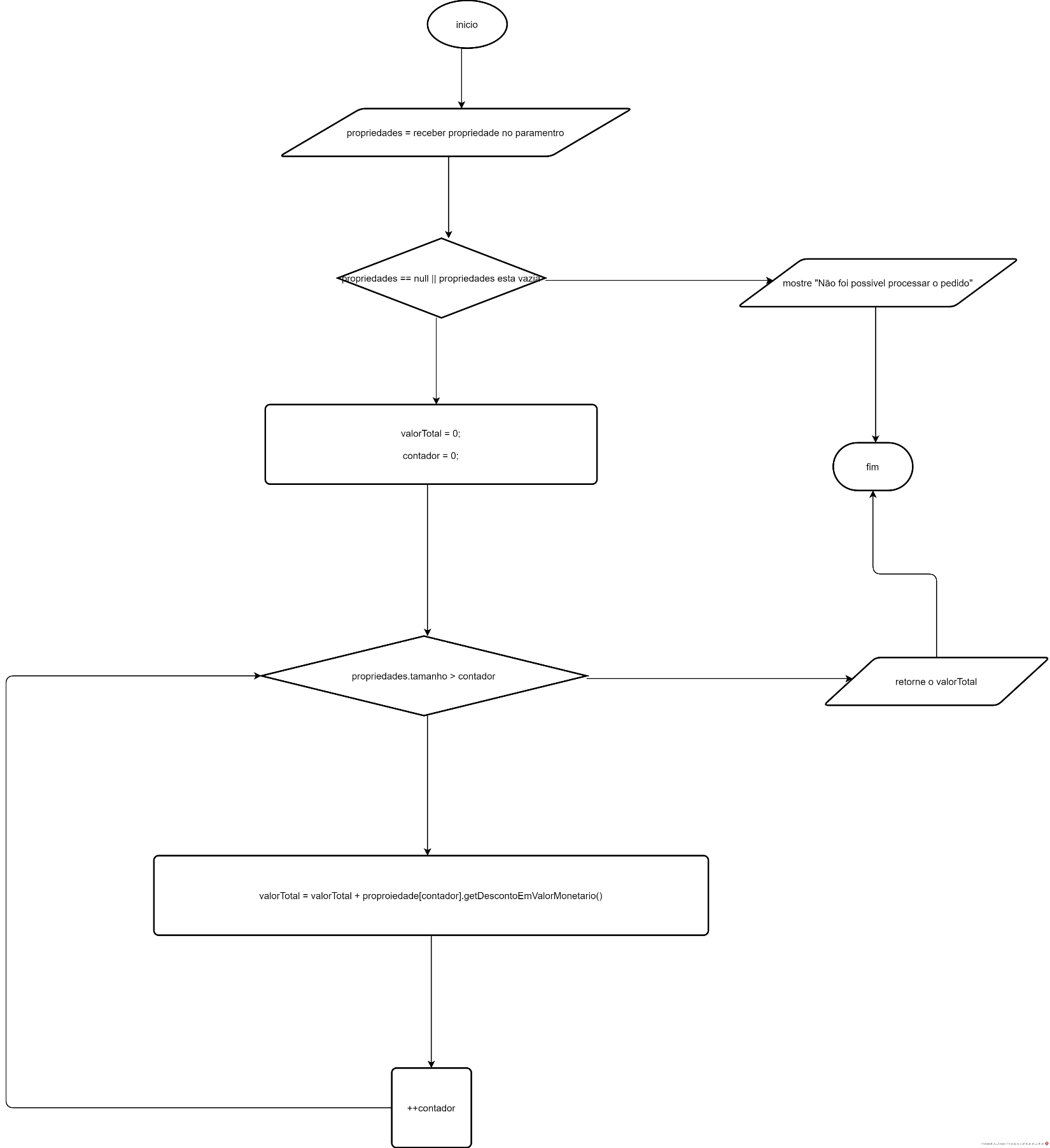
Para fazer está funcionalidade, reutilizei uma outra que havia feito, á que me trás o lucro total recebido, assim só é remover o 70% do lucro e já está.



##### Calcular o valor total dos descontos, sabendo que em todas flats do tipo 3 existe um desconto de 10% e dotipo 2 existe um desconto de 5% sobre o valor da propriedade.

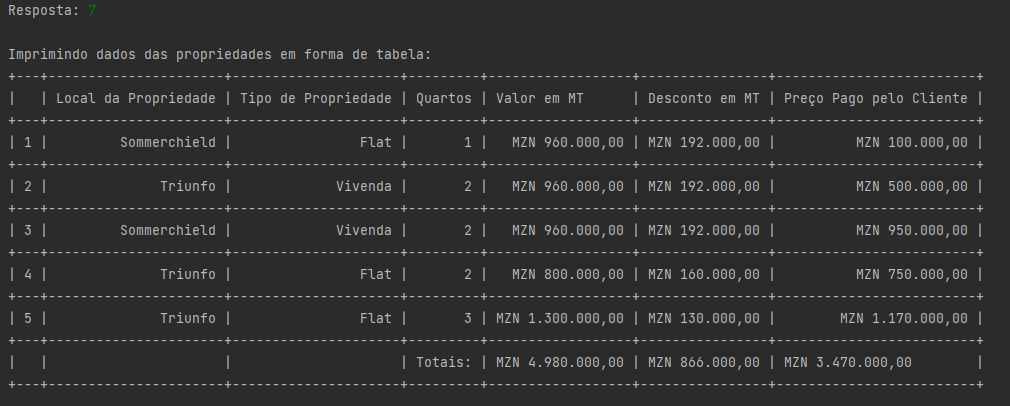
Para fazer esta funcionalidade tive que mudar a estrutura, agora passamo a ter um campo na classe propriedade para armazenar o desconto em percentagem, o utilizador pode inserir mas caso não insira ele vai usar o valor padrão.

Assim posso em qualquer momento somar esses descontos. Mas para conseguir isso tipode adicionar um método que retornasse o valor de descontos em valores motentarios.



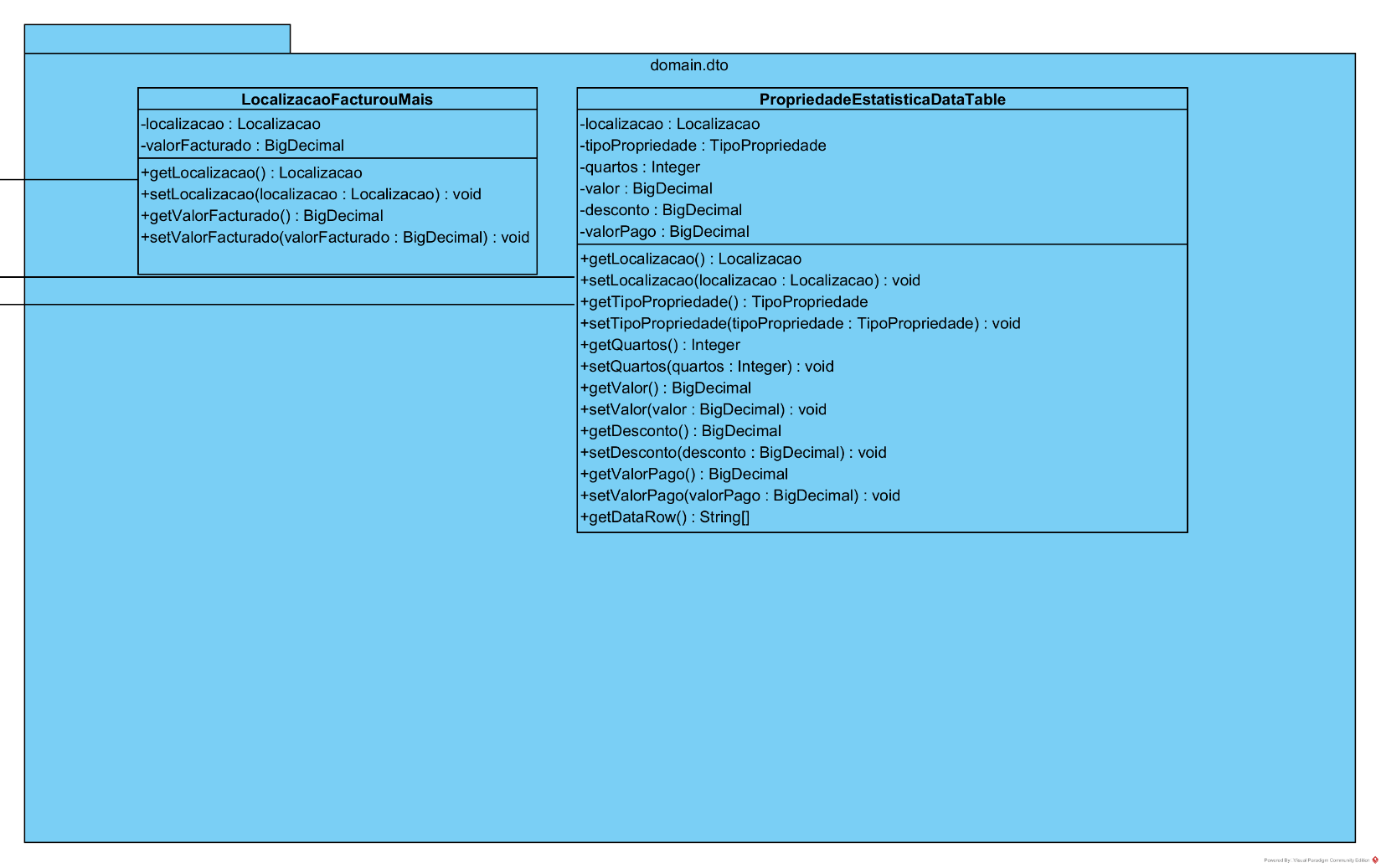
##### visualizar os dados lidos e calculados na forma de tabela com cabeçalho de seguinte maneira (exemplo):

Para conseguir imprimir os dados em uma tabela tivemos que fazer o uso de uma biblioteca chamada Ascii-table essa biblioteca nos permitir imprimir dados com essa estrutura:



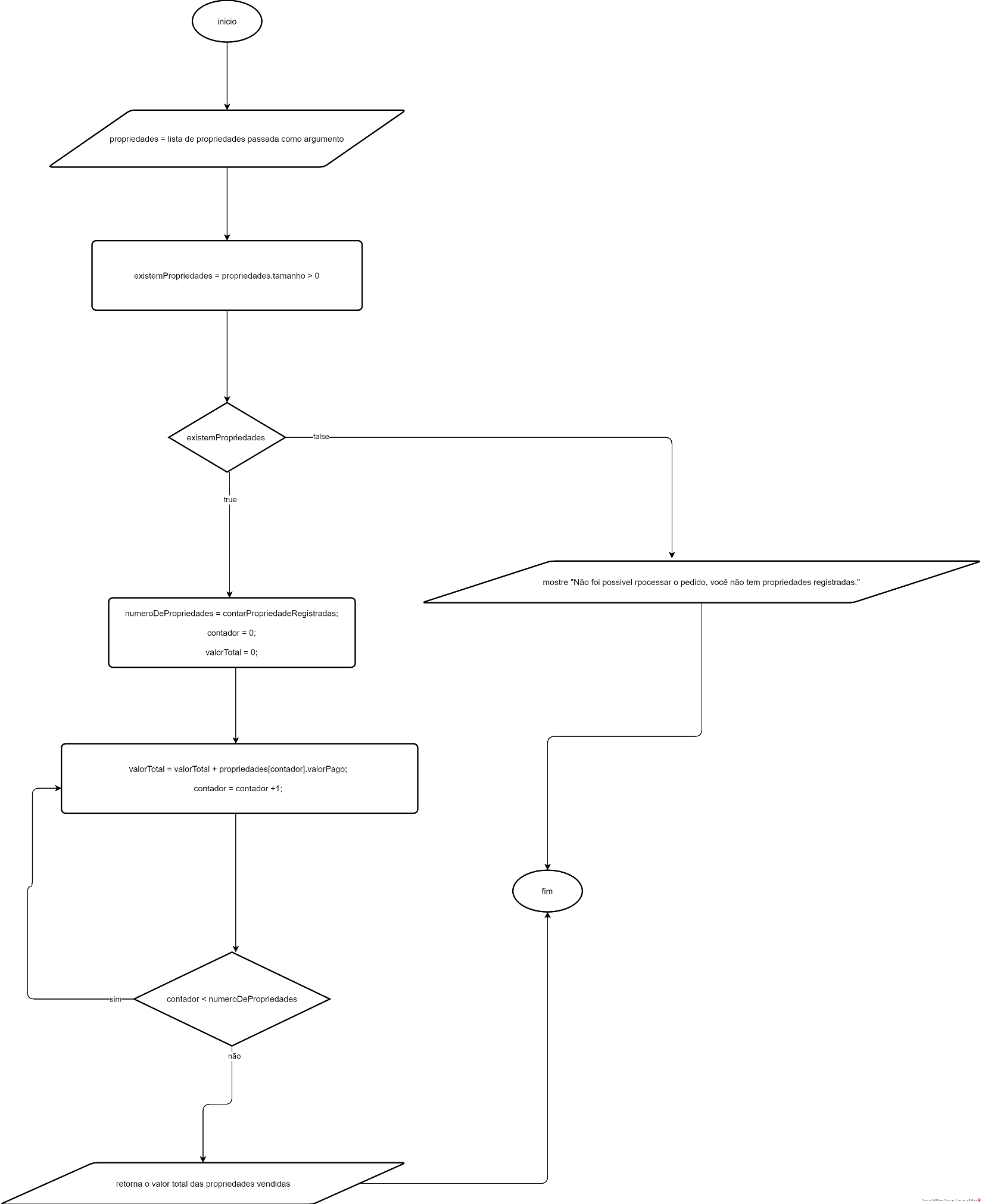
Tive também que fazer o uso de DTOs, classe que servem simplesmente para transferir dados elas não tem valor no domínio do problema.

Assim ficou fácil, veja os diagramas:

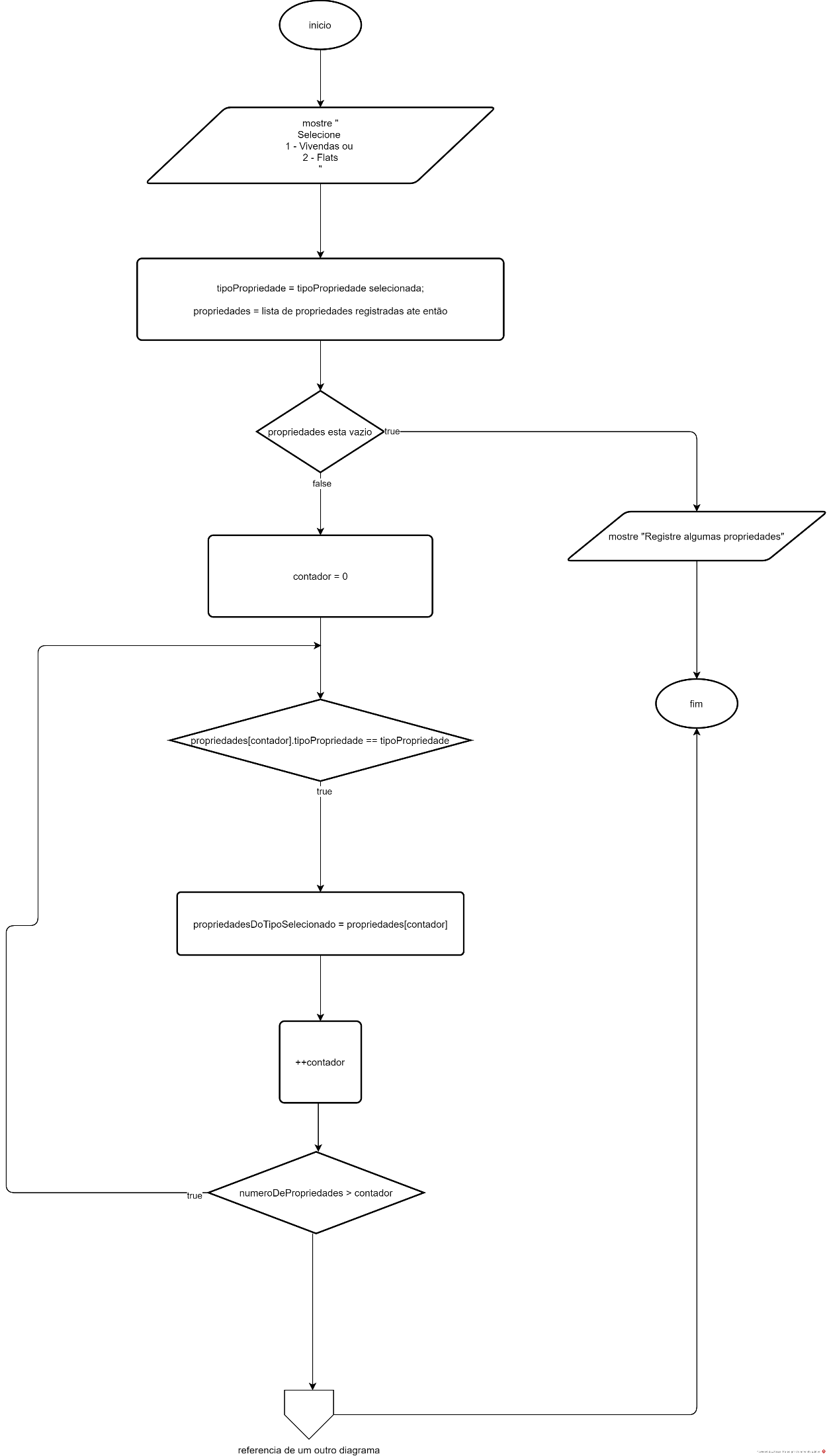


##### Calcular o valor total recebido em vivendas e flats.

Para resolver este problema tive que criar uma método mais geral que fizesse o trabalho para mim mais fácil, apos isso criei um especializado que recebe o tipo de propriedade e depois cria a lista de propriedades e usa o método mais grão, olha os diagramas:



Agora veja o método mais especializado:



# Conclusão

Aprendemos muito com o desenvolvimento desta plataforma, aprendemos a trabalhar de forma eficiente em uma aplicação que roda no console.

Percebi que posso fazer mais o uso de Orientação a objecto e criar aplicações mais abertas para serem feitas manutenção.