

análise e desenvolvimento de sistemas

tiago moreira pimentel

projeto integrado ii

**1º Semestre**

Cajamar

2022

tiago moreira pimentel

projeto integrado ii

**1º Semestre**

Produção Textual Referente ao Projeto Integrado Interdisciplinar do 1º Semestre do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, apresentado como requisito parcial para a obtenção de média semestral na disciplina de:

* Projeto Integrado - I

Orientador: Prof. Esp. Eduardo Viana de Almeida

Professores Titulares:

* Profª. Ms. Adriane Aparecida Loper
* Profª. Ms. Vanessa M. Leite
* Prof. Ms Gilberto Fernandes Junior
* Prof. Ms. Dorival Magro Junior

Cajamar

2022

SUMÁRIO

[**1 INTRODUÇÃO 1**](#_Toc172266842)

[**2 DESENVOLVIMENTO 1**](#_Toc172266843)

[2.1 Lógica Computacional 1](#_Toc172266844)

[2.2 Engenharia de Software 1](#_Toc172266846)

[2.3 Modelagem de Dados 1](#_Toc172266847)

2.4 Algoritmo e Programação Estruturada.................................1

2.5 Análise e Modelagem de Sistemas......................................1

[**3 CONCLUSÃO 1**](#_Toc172266848)

**REFERÊNCIAS 1**

# INTRODUÇÃO

O presente trabalho tem como objetivo colocar em pratica todo o conhecimento e aprendizado obtidos nos últimos meses no decorrer do 1º Semestre de Análise e Desenvolvimento de Sistemas., abordando situações problemas especificas de cada matéria, sendo elas Logica Computacional, Engenharia de Software, Modelagem de Dados, Algoritmo e Programação Estruturada e Análise e Modelagem de Sistemas.

As situações problemas são importantes para consolidar o aprendizado, e cada matéria possui uma situação especifica que pode nos ajudar a raciocinar e tomar as nossas decisões diárias com mais tranquilidade e precisão.

# DESENVOLVIMENTO

Iniciamos este trabalho com uma situação problema da matéria de Logica Computacional, lecionado pela Prof. Adriane Aparecida Loper que nos ensinou passo a passo como utilizar as ferramentas necessárias para a resolução deste exercício.

Em seguida teremos uma atividade da matéria de Engenharia de Software, lecionado pela Prof. Vanessa M. Leite sobre o Desenvolvimento Orientado a Testes e sua importância no desenvolvimento dos nossos projetos.

Logo após iremos exercitar os nossos conhecimentos com estudos sobre o desenvolvimento de Bancos de Dados com uma atividade da matéria Modelagem de Dados, lecionado pelo Prof. Marco Ikuro Hisatomi sobre a criação do MER e a utilização das ferramentas CASEs.

Em Algoritmos e programação estruturada também lecionado pela Prof. Vanessa M. Leite iremos desenvolver um programa simples em linguagem de programação C.

E por fim realizaremos uma atividade da matéria de Análise e modelagem de sistemas lecionado pela Prof. Vanessa Matias Leite, onde abordaremos assuntos sobre o diagrama de classes da UML, como utilizar e suas importâncias.

## **Lógica computacional**

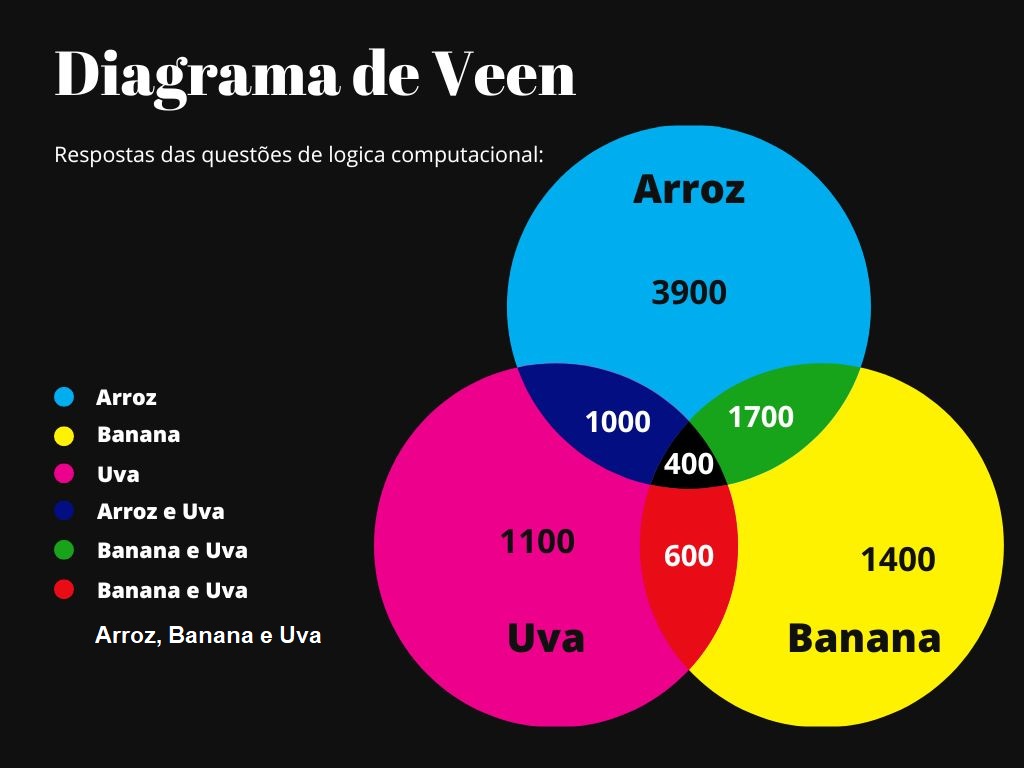
Utilizando o Diagrama de veen que desenvolvi podemos analisar que:

A) 6400 agricultores plantam uma das três espécies.

B) 3400 **agricultores não plantam nenhuma das três espécies.**

C) 5300 **agricultores plantam arroz ou banana, mas não plantam uvas.**

D) 1.100 **agricultores plantam apenas uvas.**



## **engenharia de software**

O desenvolvimento orientado a testes – TDD, o seu próprio nome já nos dá uma boa ideia do que ele exatamente é, significa entender os obstáculos que precisam ser superados, e até mesmo um teste em que você precisa passar antes mesmo de escrever a primeira linha de um código, desta forma seu software é desenvolvido da melhor forma possível com as soluções adequadas.

Este processo funciona da seguinte forma, no TDD ao implementar ou desenvolver novas funcionalidades ao nosso software devemos criar diversos testes que provavelmente vão apresentar falhas, e então aplicamos a solução superando os desafios e trazendo mais qualidade para a nossa funcionalidade, para depois desenvolver o código.

O TDD possui cinco etapas fundamentais, sendo elas:

• Criação de um código para testar o resultado de uma nova função.

• Aplica o teste ao produto para identificar uma falha, cria uma estratégia para ter a resolução.

• Cria uma função, programa ou iteração com foco em passar nas etapas sem preocupações.

• Testa repetidas vezes até ser aprovada).

• Escreve a versão final do software com qualidade.

Só de entender como esse processo funciona, já ficam claras as vantagens que ele traz para um desenvolvedor, principalmente se ele trabalha por conta própria com seu produto, resumindo tudo isso, o Desenvolvimento Orientado a Testes é um conceito de programação que organiza melhor o seu trabalho e permite que você ofereça produtos com muito mais qualidade em muito menos tempo.

## **modelagem de dados**

Podemos utilizar a ferramenta **DiagramsNet para elaborar um modelo Entidade Relacionamento (MER), sua interface de usuário pode ser utilizada para criar diagramas como:**

* **Fluxogramas**
* **Wireframes**
* **Diagramas UML**
* **Organogramas**
* **Diagramas de rede**

**O MER é utilizado para desenvolver e modelar bancos de dados relacionais, O grau de relacionamento é unário, binário, ternário, quadrinário e n-ário e os graus de cardinalidade são:** 1-1 (um para um), 1-N (um para muitos), N-1 (muitos pare um) e N-N (muitos para muitos).

Um dos benefícios de se utilizar este modelo de desenvolvimento é a facilidade de criar se diagramas para nosso banco de dados e também documentar o sistema que está sendo modelado.

## **Algoritmo e programação estruturada**

Como podemos ver na imagem 1, desenvolvi um código utilizando a IDE Dev.++, que armazena 3 informações nas devidas variáveis do tipo char, onde o usuário devera inserir os dados Nome, Endereço e Telefone. E em seguida como podemos analisar na imagem 2, o compilador imprime na tela as informações armazenadas nas variáveis.

Imagem 1

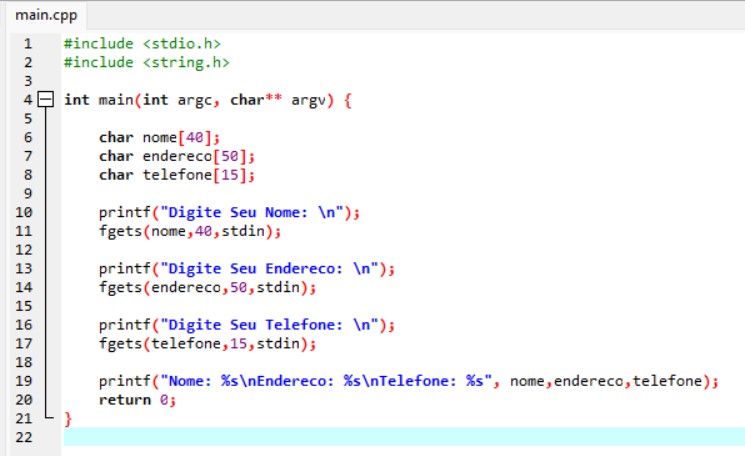
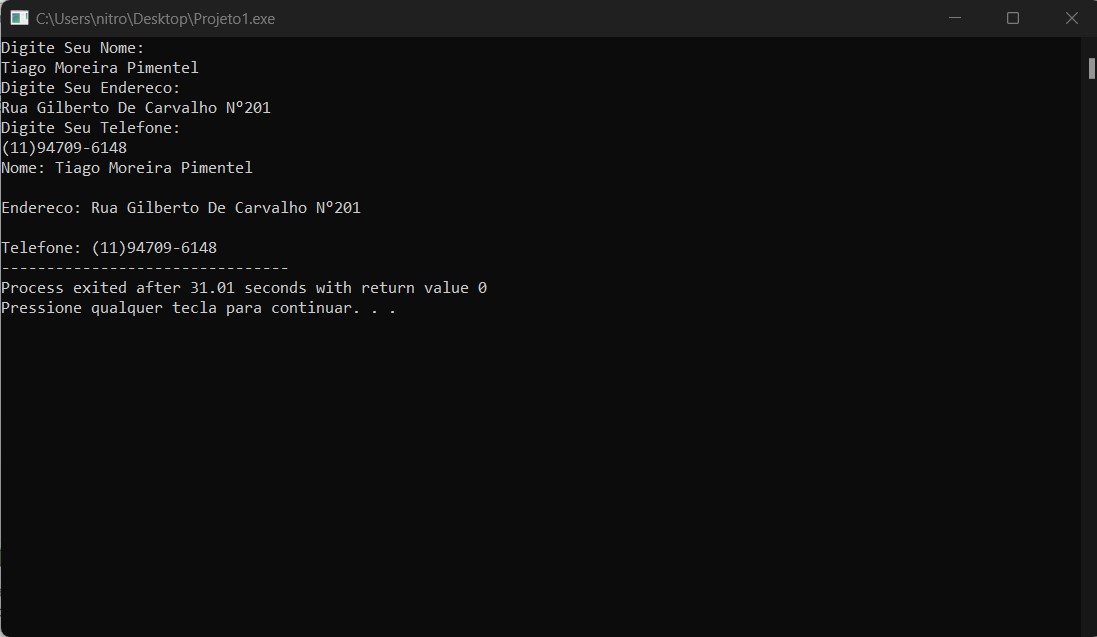


Imagem 2



## **Análise e modelagem de sistemas**

O encapsulamento faz a união de partes isoladas de um software, e podemos acessa-las separadamente, na programação orientada a objeto o encapsulamento tem o poder de deixar oculto ao usuário as ações executadas pela classe em questão.

A herança é um conceito orientado a objetos que permite que as classes compartilhem propriedades e métodos, é utilizada para ganhar tempo e otimizar os códigos.

A agregação acontece quando existe dependência entre os relacionamentos, quando precisamos associar um fato a outro fato, uma classe ‘filha’ pode existir independente de uma classe ‘pai’.

As composições acontecem quando uma classe está contida dentro de outra classe, por tanto os objetos da classe estão associados e dependem um do outro.

# Conclusão

Este trabalho me proporcionou maior conhecimento nas diversas matérias contidas neste projeto, desafiando a minha capacidade de criação e desenvolvimento gráfico e textual, ao realizar pesquisas em diversos sites e também nos materiais de apoio da instituição como por exemplo os livros únicos de cada matéria podemos evoluir os nossos pensamentos e consolidar o nosso aprendizado, e tenho certeza que cada aprendizado e cada competência adquirida durante este projeto será muito bem aplicada.

**REFERÊNCIAS BIBLIOGRÁFICAS:**

Equipe Locaweb. Entenda o que é TDD e quais são as suas vantagens. blog.locaweb. Disponível em:< [Entenda o que é TDD e quais são as suas vantagens - Blog da Locaweb](https://blog.locaweb.com.br/temas/codigo-aberto/entenda-o-que-e-tdd-e-quais-sao-as-suas-vantagens/)>. Acessado em: 24/10/2022.