



Sistema de Ensino Presencial Conectado

Superior em Tecnologia em Análise e desenvolvimento de sistemas

ALUNO: rodrigo gotardo

Trabalho apresentado ao Curso Superior de Tecnologia em Análise e Desenvolvimento de sistemas:  **SISTEMA ZER@DENGUE**

NOME DO AUTOR:

RODRIGO GOTARDO

**ANÁLISE E DESENVOLVIMENTO DE SISTEMA**

Trabalho interdisciplinar apresentado à Universidade

Norte do Paraná (Unopar), como requisito parcial

para obtenção da media semestral na disciplina de :

Análise Orientada a Objeto l, Banco de Dados, Ed-Responsabilidade

Social, Linguagens de Programação e Estrutura de Dados,

Organização de Computadores

Tutor eletrônico:

Tiago Barros dos Santos

SUMÁRIO

[1 INTRODUÇÃO 3](#_Toc34473737)

[2 DESENVOLVIMENTO. 4](#_Toc34473738)

[3 Análise Orientada a Objetos I 4](#_Toc34473739)

3.1 Diagrama Use Case.....................................................................................4

4  [Banco de Dados I 7](#_Toc34473740)

[5 Arquitetura e Organização de Computadores 9](#_Toc34473741)

[6 Linguagem de Programação e Estrutura de Dados.....................................10](#_Toc34473742)

7 [CONCLUSÃO](#_Toc34473743) 11

# 

# 1INTRODUÇÃO

Este trabalho de Portifólio foi desenvolvido com base nas matérias do 2º semestre de Análise e Desenvolvimento de Sistemas.

Essa ideia que gerou o desenvolvimento do aplicativo Zer@Dengue, veio através de uma situação fictícia que o personagem Alberto que caminhava por uma rua próxima a sua casa, e em um terreno baldio viu vários entulhos. Preocupado com a dengue, ele procurou a empresa de Desenvolvimento de Software Oráculo para desenvolver o aplicativo. Serão apresentados os diagramas, levantamento dos equipamentos e a concepção do banco de dados.

# DESENVOLVIMENTO.

## 3 Análise Orientada a Objetos I

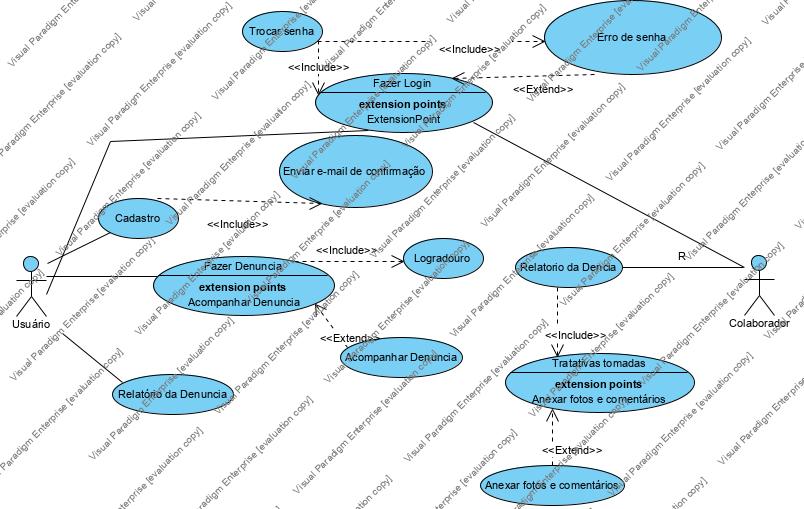
O software durante o seu desenvolvimento é dividido em estágios, conceitos abstratos, soltos comumente utilizados para dividir as atividades que seguem nas etapas do desenvolvimento, temos também o ciclo de vida de desenvolvimento de software, que vai garantir a qualidade do software como também a fácil manutenção, e o gerenciamento do projeto.

A orientação a objetos surgiu com intuito de minimizar os problemas encontrados até então na criação de softwares complexos, projetados por meio de decomposição funcional e sub-rotinas

* 1. O Diagrama de Use Cases:

O diagrama de caso de uso mostra um conjunto de casos de uso, atores e a relação entre eles. Essa relação é identificada pelo uso de linha de comunicação entre atores e casos de uso, que indicam o fluxo de informação. Sua finalidade é apresentar quais são os atores do sistema e a sua missão, sendo as responsabilidades a que o sistema a ser desenvolvido deverá atender.

Os casos de uso do sistema Zera Dengue foram colocados em uma única imagem para melhor definição, conforme segue:



**Figura 1: Diagrama de Caso de Uso**

* + 1. A documentação de todos os Use Cases, no formato numerado, com a descrição dos Cenários Principal e Alternativos:

**Use Case: Manter Zer@aDengue**

**Descrição: Cadastro de usuário no Sistema de Denúncias de Focos da Dengue.**

**Ator: Rodrigo Gotardo**

**Cenário Pirncipal:**

1. Usuário solicita cadastro no aplicativo
2. Sistema faz a confirmação através do e-mail
3. Usuário informa o logradouro da denuncia
4. Usuário insere fotos e comentários opcionais
5. Sistema valida informações
6. Sistema gera um número de protocolo
7. Usuário solicita relatório da denúncia através do número do protocolo
8. Sistema valida o protocolo
9. Sistema mostra relatório.

**Cenário Alternativo 3:**

* 1. Usuário informa estado
  2. Sistema valida estado
  3. Usuario informa cidade
  4. Sistema valida cidade
  5. Usuario valida bairro
  6. Sistema valida o bairro
  7. Usuario informa número ou referência.

**Use Case: Manter Zer@aDengue**

**Descrição: Funcionário verifica relatorio do Sistema de Denúncias de Focos da Dengue.**

**Ator: Rodrigo Gotardo**

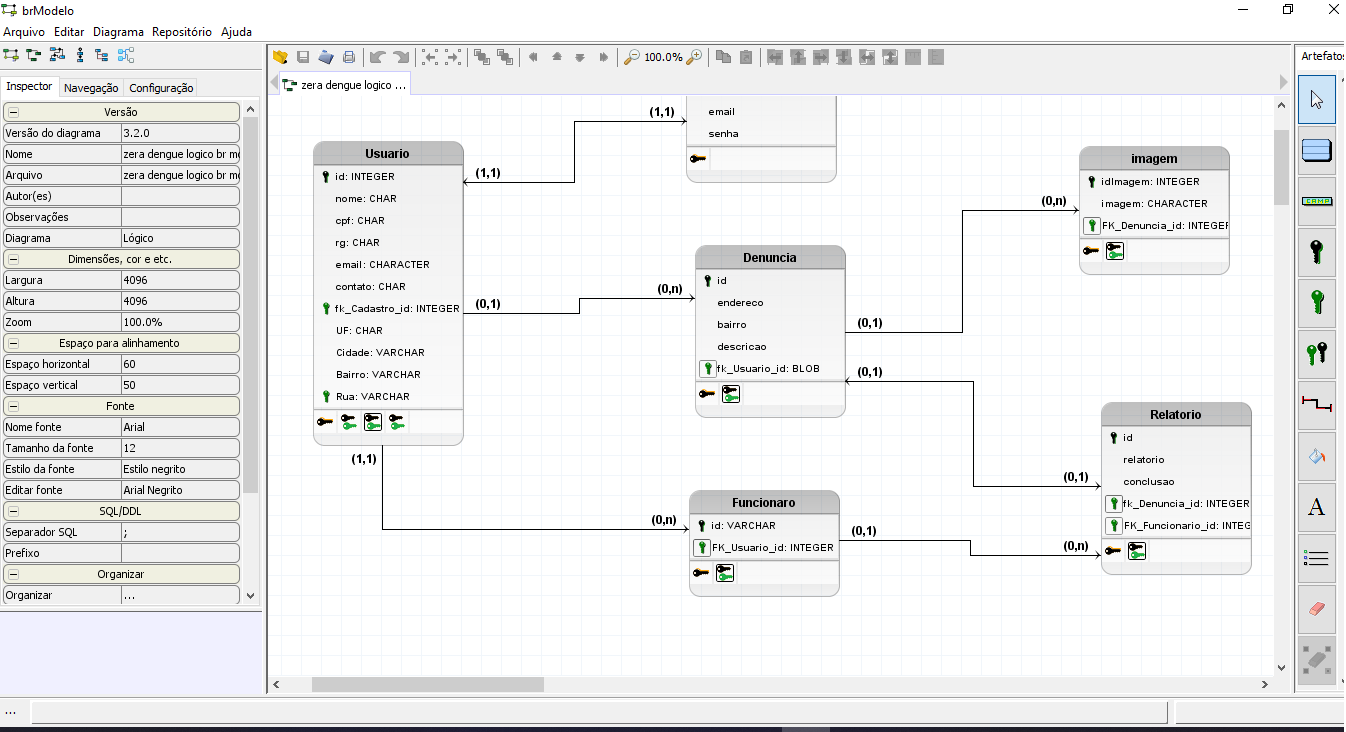
**Cenário Pirncipal:**

1. Funcionário faz login no sistema
2. Sistema valida login
3. Funcionário solicita relatório
4. Sistema verifica as denúncias realizadas e não tratadas
5. Funcionário solicita impressão do relatório
6. Sistema imprime relatório
7. Funcionário inclui tratativas da denuncia
8. Sistema envia enviar mensagem para usuário.

**Tabela 1: Descrição dos casos de uso**

* + 1. O Diagrama de Classe:

De acordo com Gedes (2007), o diagrama de classes é o mais utilizado entre os diagramas UML. Dentre seus pricipais objeticos, pode-se citar como principal, a visualização das classes utilizadas pelo sistema e como é o relacionamento entre si. O diagrama de classe passui uma visão estática, a qual é representada por classes organizadas e com uma estrutura logica definida. Ao modelar esse diagrama, cada classe é abstraída com a sua lista de atributos e métodos.



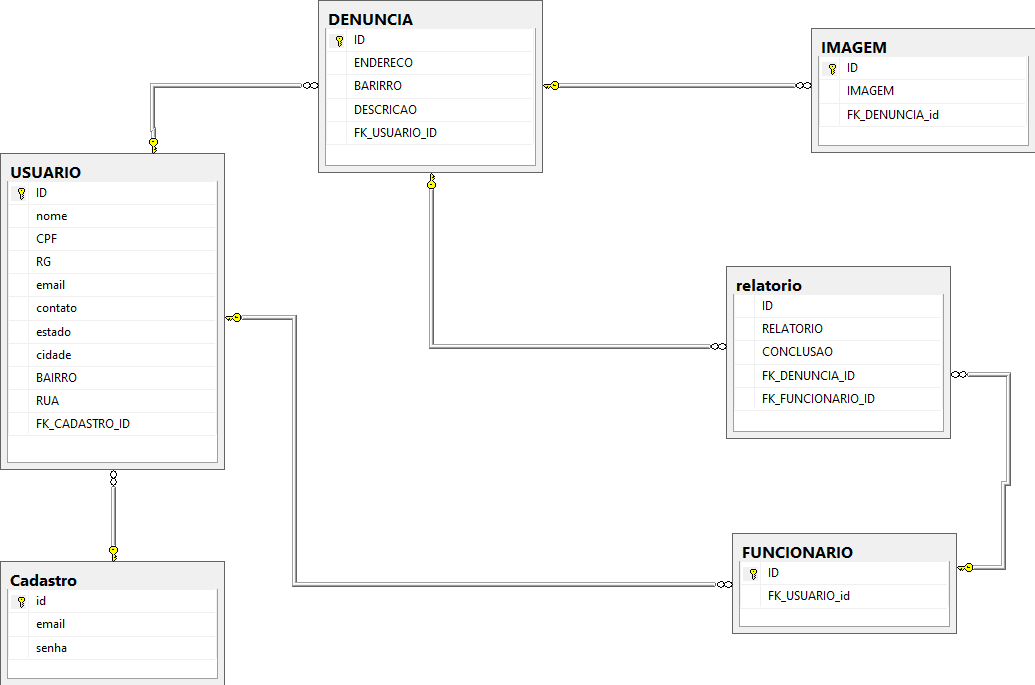
**Figura 2: Diagrama de Classe**

## Banco de Dados I

A busca pela informação, tanto no âmbito comercial como no pessoal, cresce demasiadamente a cada dia, e esse crescimento traz a necessidade de armazenamento e manipulação de dados. Toda manipulação de dados tem um único fim, gerar informações sobre oum determinado assunto ou negócio.

Para suprir tal necessidade de armazenamento e, posteriormente, de manipula-los, foi desenvolvido um banco de dados capaz de sanar essa ncessscidade.

A informação tem ocupado um destaque nas empresas, e também tem adquirido um potencial de valoração para as organizações, proteger a informação é a maneira de zelar pelo ativo da empresa, para o negocio a maximizar o retorno sobre o investimento e as oportunidades de negocio.



**Figura 3: MER**

|  |
| --- |
| use ZER@\_DENGUE  GO  CREATE TABLE Cadastro ( id INT IDENTITY(1,1) NOT NULL,  email VARCHAR(50) NOT NULL,  senha VARCHAR(12) NOT NULL,  PRIMARY KEY CLUSTERED (id) )  GO  CREATE TABLE USUARIO ( ID INT IDENTITY (1,1) NOT NULL,  nome VARCHAR (50) NOT NULL,  CPF CHAR (11) NOT NULL,  RG VARCHAR (15) NOT NULL,  email VARCHAR (50) NOT NULL,  contato VARCHAR (11),  estado CHAR (2) NOT NULL,  cidade VARCHAR (25) NOT NULL,  BAIRRO VARCHAR (50) NOT NULL,  RUA VARCHAR (50) NOT NULL,  FK\_CADASTRO\_ID INT NOT NULL,  PRIMARY KEY CLUSTERED (ID),  CONSTRAINT [FK\_USUARIO\_Cadastro\_FK\_CADASTRO\_ID] FOREIGN KEY (FK\_CADASTRO\_ID) REFERENCES Cadastro (id))  GO  --denuncia  CREATE TABLE DENUNCIA ( ID INT IDENTITY (1,1) NOT NULL,  ENDERECO VARCHAR (50) NOT NULL,  BARIRRO VARCHAR (50) NOT NULL,  DESCRICAO TEXT,  FK\_USUARIO\_ID INT NOT NULL,  PRIMARY KEY CLUSTERED (ID),  CONSTRAINT [FK\_DENUNCIA\_USUARIO\_FK\_USUARIO\_ID] FOREIGN KEY (FK\_USUARIO\_ID) REFERENCES USUARIO (ID))  GO  --imagem  CREATE TABLE IMAGEM (ID INT IDENTITY (1,1) NOT NULL,  IMAGEM IMAGE NOT NULL,  FK\_DENUNCIA\_id int not null,  PRIMARY KEY CLUSTERED (ID),  CONSTRAINT [FK\_IMAGEM\_DENUNCIA\_FK\_DENUNCIA\_ID] FOREIGN KEY (FK\_DENUNCIA\_ID) REFERENCES DENUNCIA (ID))  GO  --funcionario  CREATE TABLE FUNCIONARIO (ID INT IDENTITY (1,1) NOT NULL,  FK\_USUARIO\_id int not null,  PRIMARY KEY CLUSTERED (ID),  CONSTRAINT [FK\_FUNCIONARIO\_USUARIO\_FK\_USUARIO\_ID] FOREIGN KEY (FK\_USUARIO\_id) REFERENCES USUARIO (ID))  GO  --relatorio  CREATE TABLE relatorio (ID INT IDENTITY (1,1) NOT NULL,  RELATORIO TEXT NOT NULL,  CONCLUSAO VARCHAR NOT NULL,  FK\_DENUNCIA\_ID INT NOT NULL,  FK\_FUNCIONARIO\_ID INT NOT NULL,  CONSTRAINT [FK\_RELATORIO\_DENUNCIA\_FK\_DENUNCIA\_ID] FOREIGN KEY (FK\_DENUNCIA\_ID) REFERENCES DENUNCIA (ID),  CONSTRAINT [FK\_RELATORIO\_FUNCIONARIO\_FK\_FUNCIONARIO\_ID] FOREIGN KEY (FK\_FUNCIONARIO\_ID) REFERENCES FUNCIONARIO (ID))  GO |

**Quadro 1: script SQL**

## Arquitetura e Organização de Computadores

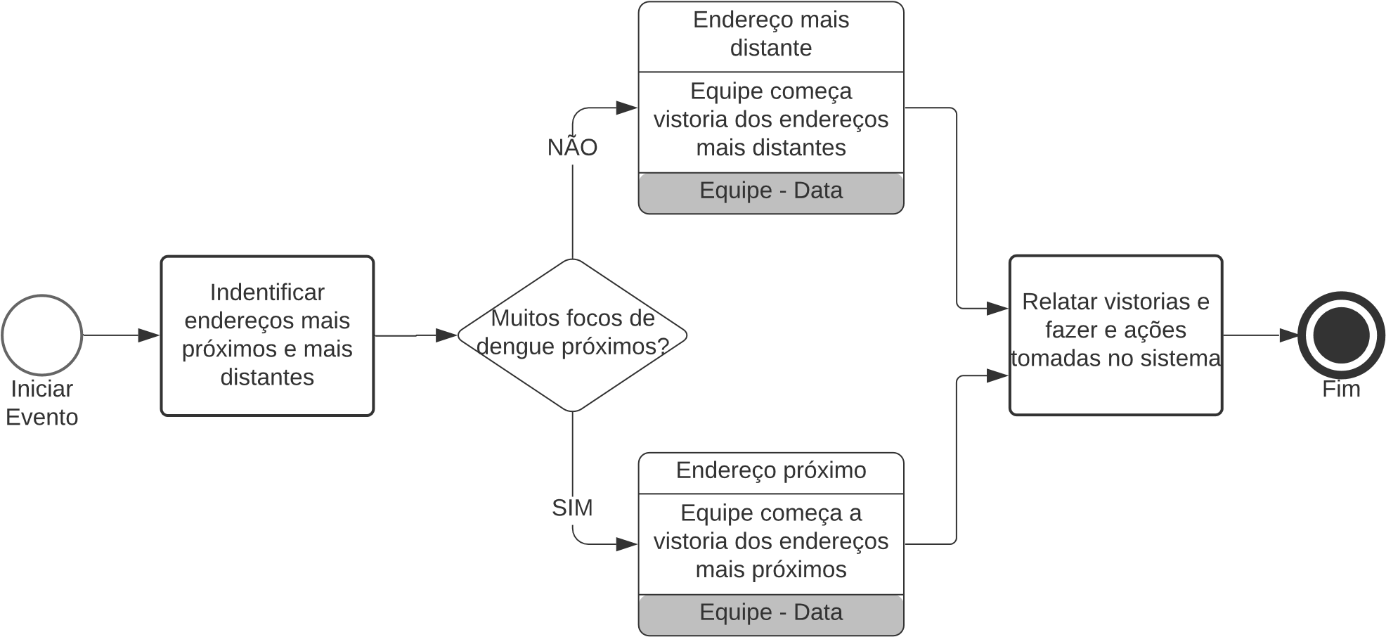
Para atender a demanda do Sistema Zer@Dengue foi selecionado um PC Desktop completo, com configurações ideais e valor acessível:

* Gabinete ATX Cougar
* Fonte 550W
* Placa -mãe Gigabyte GA-78LMT- USB3
* PROCESSADOR AMD FX-8350
* 8BG de memoria ram
* 1TB de disco rígido
* Mouse e Teclado
* Monitor Acer V6 19.5 polegadas
* Impressora Multifuncional Epson EcoTank L3150
* Total : R$ 3.652,00

## LINGUAGEM DE PROGRAMAÇÃO E ESTRUTURA DE DADOS

Para facilitar a verificação das denuncias feitas pelos usuários do Aplicativo Zer@Dengue, pelos profissionais de saúde e vigilância sanitária, foi utilizado uma das soluções mais utilizadas em lista, quando se esta tratando de uma segmentação e organização, busca binária.

A busca binária é um eficiente algoritmo para encontrar um item em uma lista ordenada de itens. Ela funciona dividindo repetidamente pela metade a porção da lista que deve conter o item, até reduzir as localizações possíveis a apenas uma.

**Figura 4: Diagrama**

# CONCLUSÃO

Depois de ter levantado todos os dados e requisitos, o para a conclusão deste PTI, foi necessário a utilização de analise orientada a objetos, como banco de dados, codando um script para o App Zer@Dengue.

Foi analisado os requisitos de softwares disponíveis no mercado agragando melhor custo beneficio, onde este fará parte dos equipamentos da empresa para rodar o programa, e a solução para o funcionário responsável por analisar as denuncias no Zer@Dengue pudesse ter os relatórios e dar a resposta da denuncia.

###### REFERÊNCIAS

FABRIS, P. P. G.; MORAES, E. M.; HISATOMI, M.. **Análise orientada a objetos I**. Londrina: Editora e Distribuidora Educacional S.A., 2018.

PRESSMAN, R.; MAXIM, B. **Engenharia de software: uma abordagem profissional**. 8 ed. Porto Alegre: AMGH, 2016. Disponível em < https://integrada.minhabiblioteca.com.br/books/9788580555349> acesso em 06 mar. 2020.

SANTANA, Gisele Alves. **Linguagens de programação e estruturas de dados**. Londrina: Editora e Distribuidora Educacional S.A. 2018.

Artigo Engenharia de Software2 -**Análise Orientada a Objetos** 15/05/2020

<<https://www.devmedia.com.br/artigo-engenharia-de-software-2-analise-orientada-a-objetos/9150>> acessado 15/05/2020

FABRIS, P. P. G.; MORAES, E. M.; HISATOMI, M.. **Banco de dados l. Londrina: Editor**a e Distribuidora Educacional S.A 2018.