*Uma imagem com texto, Tipo de letra, Gráficos, logótipo

Descrição gerada automaticamente*



**Programação orientada a objetos**

LICENCIATURA ENGENHARIA INFORMÁTICA MÉDICA

2ºano

2023/2024

Docente: Tiago Castro

Grupo 9

No.22845 – Mateus Costa

No.28027 – Houda El Manssouri

No.28040 – Beatriz Fernandes

No.29860 – Júlio Pereira

Índice

[Introdução 2](#_Toc182919792)

[Modelo de Negócio 3](#_Toc182919793)

[Objetivos 3](#_Toc182919794)

[Classes Identificadas 3](#_Toc182919795)

[Implementação Essencial das Classes 4](#_Toc182919796)

[Desafios Encontrados 7](#_Toc182919797)

Repositório GitHub com o trabalho

<https://github.com/pumata22/Trabalho_POO.git>

Introdução

Neste relatório demonstramos o nosso desenvolvimento no trabalho prático da disciplina de Programação Orientada a Objetos. No qual tentamos tentar explicar um pouco algumas partes mais importantes do nosso código.

# **Modelo de Negócio**

O Modelo de Negócio que nos decidimos adotar é o de uma aplicação de gestão de atividades de socorro.

# **Objetivos**

O nosso objetivo com este trabalho é conseguir planear, projetar uma aplicação de atividades de socorro que funcione e que possa ser utilizada.

# **Classes Identificadas**

Classes: - Classes: --> Bombeiros

--> Inem

--> ForcasArmadas

--> Ocorrencia

--> Pessoa

--> ProtecaoCivil

--> Veiculos

- Abstratas: --> AbstrataVeiculos

- Interfaces: --> IVeiculos

# **Implementação Essencial das Classes**

**Bombeiros:**

Atributos

private int idQuartel;

private string moradaQuartel;

private string areaAtuacao;

**Inem:**

Atributos

private int qtdMedicos;

private int qtdEnfermeiros;

Construtores

public Inem(int qtdMedicos, int qtdEnfermeiros) {}

**ForçasArmadas:**

Atributos

private string TipoForca;

private string moradaBase;

private string areaAtuacao;

Construtores

public ForcasArmadas(string TipoForca, string moradaBase, string areaAtuacao) {}

**Ocorrência:**

Atributos

private string morada;

private string veiculos;

private string observacoes;

private int data;

private int idade;

private int nVitimas;

private int nCodu;

private int identificadorTipoEmergencia;

private int nVeiculos;

private int coordenadas1, coordenadas2;

private int nOperacionais;

private double horaAlerta,Horafecho;

private double areaArdida;

Construtor emergências medicas e agressão

public Ocorrencia(string morada, string veiculos, string observacoes, int nVitimas,int idade,int nCodu,int identifacadorTipoEmergencia,int nOperacionais,Pessoas pessoa) {}

Construtor acidente

public Ocorrencia(string morada, string veiculos, string observacoes, int nVitimas,int nCodu, int identifacadorTipoEmergencia,int coordenadas1, int coordenadas2, int nOperacionais, Pessoas pessoa) {}

Construtor incendio urbano e industrial

public Ocorrencia(string morada, string veiculos, string observacoes, int nVitimas, int identifacadorTipoEmergencia, int coordenadas1, int coordenadas2, int nOperacionais) {}

**Pessoa:**

Atributos

private int nOperacional;

private string nome;

private int idade;

private int moradaPessoa;

Construtores

public Pessoas(int nOperacionalPar, string nomePar, int idadePar, int moradaPessoaPar) {}

**Veículos:**

Atributos

private string sigla;

private string categoria;

private int lugares;

Construtores

public Veiculos(string siglaPar, string categoriaPar, int lugaresPar) {}

**Proteção civil:**

Atributos

private int IdBasePC;

private string moradaBase;

private string area;

Construtores

public ProtecaoCivil(int IdBasePC\_par, string moradaBase\_par, string area\_par, Veiculos veiculo\_par) { }

# **Desafios Encontrados**

* Como armazenar as ocorrências, para posteriormente mostrar, ainda não entendemos o conceito total mas achamos que estamos num bom caminho;