

Relatório de trabalho prático

Infetados numa

Situação de Saúde Pública

João Ricardo João Rodrigues

Aluno nº 18845 Aluno nº 19431

Trabalho realizado sob a orientação de:

Luís Ferreira

**Linguagens de Programação II**

**Licenciatura em Engenharia de Sistemas Informáticos**

Barcelos, ABRIL de 2020

Índice

1 INTRODUÇÃO 1

1.1 Contextualização 1

1.1.1 MOTIVAÇÃO E OBJETIVOS 1

1.1.2 ESTRUTURA DO DOCUMENTO 2

2 ESTADO DA ARTE 3

2.1 Paradigma Orientado a Objetos (POO) 3

2.2 O que são Classes e Objetos 4

2.2 Pilares do POO 5

2.2.1 Abstração 5

2.2.2 Encapsulamento 5

2.2.3 Herança 6

2.2.4 Polimorfismo 7

3 METODOLOGIA 9

3.1 Descrição do Problema e como é Resolvido 9

3.2 Classes Envolvidas e Respetivo Diagrama 10/11

4 CONCLUSÃO 13

BIBLIOGRAFIA 15

ANEXOS 17

# INTRODUÇÃO

## CONTEXTUALIZAÇÃO

Este documento foi realizado no contexto da unidade curricular Linguagem de Programação II,do curso de Engenharia Informática, da Escola Superior de Tecnologia, do Instituto Politécnico Cávado do Ave.

Neste trabalho prático pretende-se criar soluções em C# para problemas reais, soluções essas capazes de resolver, ou de auxiliar o Ser Humano no seu quotidiano nas suas tarefas.

O Tema ao qual deve ser abrangido um conjunto de requisitos é o desenvolvimento de um Sistema que permita gerir pessoas Infetadas numa situação de Saúde Pública.

### MOTIVAÇÃO E OBJETIVOS

Após recolha da informação necessária para a realização do projeto, definiram-se que os

objetivos da solução informática a desenvolver deve ter em consideração tópicos bastante relevantes como:

* Implementação de uma solução com base no POO ( Paradigma Orientado a Objetos);
* Produção de um código de acordo com a norma CLS, bem documentado;
* Gerar API com a documentação do código que produzido;
* Apliacação dos pilares da POO;
* Estruturar devidamente a solução em bibliotecas de classes;

### ESTRUTURA DO DOCUMENTO

Este documento é constituído por 4 capítulos:

No **Capítulo 1. Introdução**, onde temos uma breve introdução, motivação,objetivos, e a estrutura do documento.

No **Capítulo 2.Estado da Arte**, onde é comentado após uma breve pesquisa, o que

podemos encontrar acerca do Paradigma Orientado a Objetos.

No **Capítulo 3.Metodologia**, onde é descrito a forma como é resolvido o projeto e especificado as classes envolvidas e o respetivo Diagrama de Classes.

No **Capítulo 4. Conclusão**, tem a finalidade de retirar conclusões

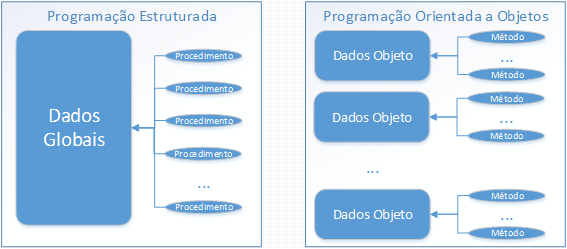
do que foi realizado no projeto, e também as aprendizagens retiradas deste mesmo.

|  |
| --- |
|  |
|  |

# ESTADO DA ARTE

## PARADIGMA ORIENTADO A OBJETOS

Programação orientada a objetos é um paradigma de programação baseado no conceito de "objetos", que podem conter dados na forma de campos, também conhecidos como atributos, e códigos, na forma de procedimentos, também conhecidos como métodos.

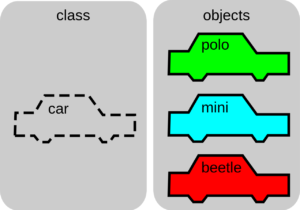




## O que são Classes e Objetos

Classes são um conjunto de características(atributos) e comportamentos que um objeto possui.Assim possuímos inúmeros exemplos capazes de explicar melhor este tipo de situação.

Consideremos um carro, este é um objeto e possui coisas muito similares com muitos outros como quatro rodas, volante, faróis, retrovisores, entre outros. Apesar de este ser único pois possui um registro só dele, podem no entanto existir outros com exatamente os mesmo atributos, semelhantes, ou totalmente diferentes ou seja podemos afirmar que este carro que estamos a falar pretence a um classe Carros.



## Pilares do POO

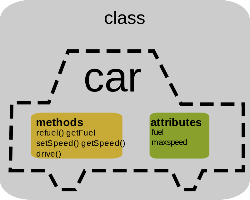
### Abstração

A abstração é um tentativa de “esconder” o que este por trás de algo, ou seja a complexidade um um problema é abstraída, tornando só visível a parte que interressa tornar pública, detalhes que não são importantes não são divulgados.É implementada através do **PUBLIC/PRIVATE** em atributos e métodos.

### Encapsulamento

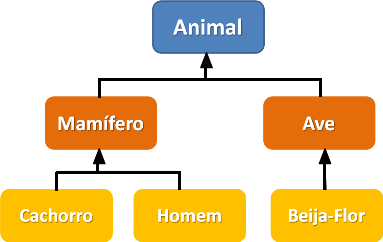
O Enapsulamento protege o estado de acessos indevidos, cada objeto tem o seu próprio estado e o seu comportamento e os atributos e métodos estão salvaguardados no interior do objeto.

O Estado não deve ser acedido diretamente, é feito através de métodos disponibilizados para esse efeito (interface pública), tendo em conta que o estado de ser sempre **PRIVADO**.



### Herança

A herança é um princípio próprio à programação orientada a objetos (POO) que permite criar uma nova classe a partir de uma já existente. Contém atributos e métodos da classe primária (da qual deriva). A principal vantagem da herança é a capacidade para definir novos atributos e métodos para a subclasse, que se somam aos atributos e métodos herdados.

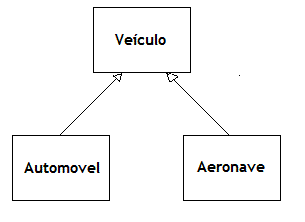


### Polimorfismo

Classes derivadas de uma mesma superclasse podem invocar métodos que têm a mesma identificação (assinatura) mas comportamentos distintos.

Aplica-se em :

* Redefinição de Métodos;
* Métodos abstratos;



# METODOLOGIA

## DESCRIÇÃO DO PROBLEMA E COMO É RESOLVIDO

Este projeto foi concebido para auxiliar “algo” seja um País, Distritos ou mesmo Munícipos numa situação de crise pública, tendo por base a utilização dos pilares do Paradigma Orientado a Objetos.

A solução pensada foi a utilização de bibliotecas para gerir diferentes elementos e a Estrutura de Dados a utilizar o ARRAY, como é o caso dos Infetados, Tratadores e Pessoas e o Main do programa onde se processa todas as funções.

Esta solução possui três biblioteca, são essa a biblioteca Pessoa, Infetado e Tratadores.

A função da biblioteca Pessoa é gerir um grupo de pessoas que por sua vez algumas ou até mesmo todas possam ser Infetados, transmitindo assim um dos pilares do POO, a herança.

A função da biblioteca Infetados é gerir um grupo de Infetados, que são pessoas, e dar todo o tipo de informações sobre um determinado indivíduo, como por exemplo registar uma pessoa como sendo infetada.Assim esta biblioteca possui uma grande importância neste projeto possuindo funções de todo o tipo como registar casos infetados, contabilizar casos, mostrar os casos por região, idade, sexo...

A função da biblioteca Tratadores é gerir um grupo de tratadores que cuidam dos infetados, ou seja que estão responsavéis por dar todo o tipo de apoio seja psicológico ou mesmo físico.

## CLASSES ENVOLVIDAS E RESPETIVO DIAGRAMA

Neste projeto estão envolvidos seis classes:

Na biblioteca Pessoa possui duas classes, a Pessoa e Pessoas.Na classe Pessoa possui os atributos cartao de cidadão, data de nascimento, idade, morada, município, nome, género e cada atributo tem uma propriedade capaz de modificar a variável e de retornar o resultado caso seja necessário.Para além disso, o construtor desta classe tem duas variações os dados por omissão e com dados vindo do exterior.

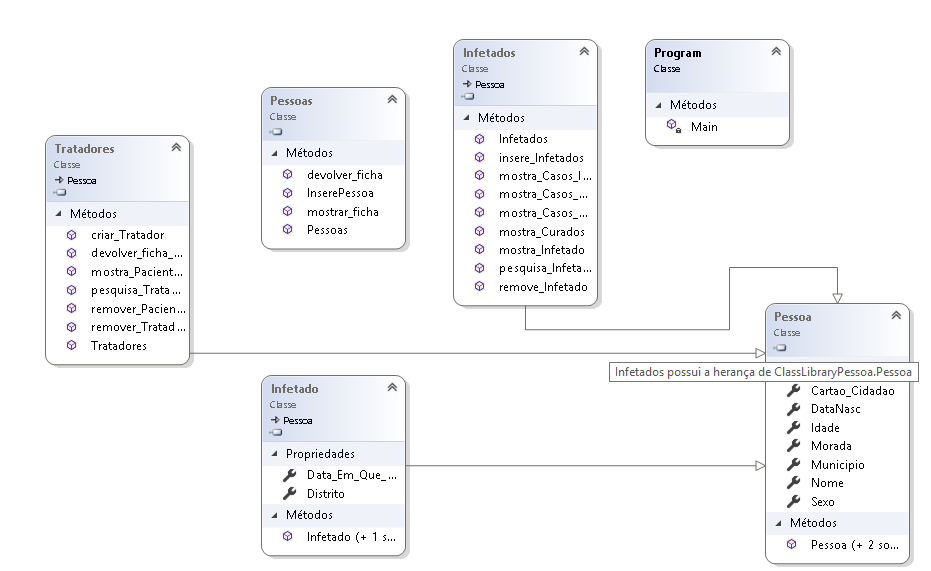
Por outro lado a classe Pessoas dentro da biblioteca Pessoa, esta gere um conjunto de pessoa, com um conjunto de métodos capaz de manipular um array de Pessoas.

Na biblioteca Infetados possui duas classes, a Infetado e Infetados.Na classe Infetado os atributos de um infetado são o distrito e data em que foi infetado, no que toca à idade, nome, nif é utilizado a biblioteca Pessoa através de uma herança.

Por outro lado a classe Infetados dentro da biblioteca Infetados, esta gere um conjunto de infetados, com um conjunto de métodos capaz de manipular um array de Infetados.

Na biblioteca Tratadores possui duas classes, a Tratador e Tratadores.Na classe Tratador os atributos de um tratador são o distrito do tratador, no que toca aos restantes atributos são manipulados pela biblioteca Pessoa através de uma herança.

Por outro lado a classe Tratadores dentro da biblioteca Tratador, esta gere um conjunto de Tratadores, com um conjunto de métodos capaz de manipular um array de Tratadores e um array de strings capaz de guardar os nif dos pacientes que vai tratar.



# Conclusão

Na nossa opinião,foi muito interessante o desenvolvimento deste projeto, deu para

Potenciar a experiência do desenvolvimento de software, Assimilar o Conteúdo da Unidade Curricular, Desenvolver Capacidades de programação em C#, Consolidar Conceitos basilares do Paradigma Orientado a Objetos (POO) e Análise de problemas reais. Durante a execução do projeto foram encontrados algumas dificuldades como a escolha da estrutura de dados para armazenar a informação e a melhor maneira de a utilizar para a solução pretendida.

Sentimos que agora estamos mais preparado para futuros projetos que nos sejam apresentados, pois este acabou por ser bastante exigente o que fez com que tivesse-mos de nos aplicar e melhorar as nossas capacidades.

Com este trabalho adquiri inúmera valias que me seram úteis para futuros projetos.

Em suma, abordamos neste trabalho todos os assuntos lecionados nas aulas e graças a isso conseguimos cumprir os objetivos propostos.

# Bibliografia

https://www.devmedia.com.br/os-4-pilares-da-programacao-orientada-a-objetos/9264

# Anexos

**REPOSITÓRIO GIT HUB:**

https://github.com/RicardoPinto18845/-18845\_19431\_LP2.git