

Relatório de trabalho prático

Sistema de gestão de uma urgência em C#

Ana Rita SILVA E André Felicidade

Aluno nº 19544 e 18861

Trabalho realizado sob a orientação de:

Luís Ferreira

**Linguagens de Programação II**

**Licenciatura em Engenharia de Sistemas Informáticos**

Barcelos, Maio de 2020

Índice

[1 Introdução 1](#_Toc38819654)

[1.1 Motivação 1](#_Toc38819655)

[1.2 Objetivos 1](#_Toc38819656)

[1.3 Estrutura do Documento 2](#_Toc38819657)

[2 Análise do problema 3](#_Toc38819658)

[3 Análise do problema pelo paradigma orientado ao objeto 4](#_Toc38819659)

[4 Repositório 6](#_Toc38819660)

[5 Problemas encontrados 8](#_Toc38819661)

[Figura 1-Herança de Classes 5](file:///C:\Users\ritaf\Downloads\Template%20relatório%20LP2%20(1)%20(1).docx#_Toc38818471)

[Figura 2- Classes Hospital,Triagem,ProcessoClinico,ConsultaMedica e Urgencia 7](#_Toc38818472)

# Introdução

## Motivação

No enquadramento da disciplina de Linguagem de Programação II, da Licenciatura em Sistemas Informáticos, pretende-se que em C#, desenvolvamos uma solução, para um sistema que nos permita gerir uma urgência hospitalar. Para que consigamos chegar a uma solução teremos de analisar o problema numa situação real, e transpor a sua solução para C#, tendo em consideração os conceitos do Paradigma Orientado a Objetos. Deste modo, começaremos por identificar e implementar classes e objetos no contexto do problema.

## Objetivos

* Analisar o problema e estruturar uma solução por camadas;
* Implementar um sistema que permita a gestão de um serviço de urgência;
* Desenvolver competências no Paradigma Orientado a Objetos;
* Identificar e Implementar classes e objetos;
* Gerar a API com a documentação do código que produzimos;
* Produzir Código de acordo com a norma CLS;
* Estruturar uma solução em bibliotecas de classes;

## Estrutura do Documento

Analise do problema real;

Estruturação da solução em camadas, pelo paradigma orientado a objetos:

– Implementação essencial das classes/diagrama de classes;

– Definição das estruturas de dados a utilizar;

– Separação por camadas das classes.

# Análise do problema

Para desenvolvermos uma solução para a criação de um sistema de gestão de um serviço de Urgência, temos que inicialmente ter em consideração o funcionamento habitual deste serviço. Quando um paciente recorre a uma Urgência, inicialmente efetua o seu registo na receção, onde são pedidos os seus dados identificativos. Posteriormente aguardam pela chamada do serviço de triagem, onde são avaliados pela equipa de Enfermeiros, que recolhem a anamnese do paciente, temperatura corporal e a tensão arterial e, atribuem uma pulseira que identifica com que nível de urgência têm de ser atendidos. O sistema de cores da pulseira atribuída estabelece o período tempo em que numa situação ideal deveriam ser atendidos. Existem 6 cores de pulseiras que podem ser atribuídas:

* Vermelho (emergente): o doente é imediatamente atendido;
* Laranja (muito urgente): o doente passa para uma sala de espera interna, em que deverá esperar pelo atendimento no máx.10 min;
* Amarela (urgente): o doente aguarda na sala de espera, devendo ser atendido no período máximo de uma hora;
* Verde (pouco urgente): o doente aguarda na sala de espera, e será atendido quando não houver doentes com atribuição de pulseira mais urgente. Deverá ser atendido em 2 horas;
* Azul (não urgente): o tempo de espera poderá ser até 4 horas;

A ordem pela qual é efetuada a chamada para o atendimento pela equipa médica, é então feita pela gestão das pulseiras atribuídas. Neste trabalho iremos assumir que depois do paciente ir ao médico, que o seu caso clínico fica resolvido, não havendo a necessidade de reencaminhamento para diferentes especialidades.

# Análise do problema pelo paradigma orientado ao objeto

Para a estruturação da resolução em camadas do problema, começamos por criar as seguintes classes:

* Pessoa;
* Utente;
* Enfermeiro;
* Medico;

Asseguramos que as classes Enfermeiro, Medico e Utente *são* uma Pessoa, através do recurso ao conceito de Herança. Deste modo as classes” *filhos”* recebem toda a informação contida na classe *“pai”.* Para facilitar a visualização deste conceito apresentamos de seguida o diagrama de classes, fig.1.

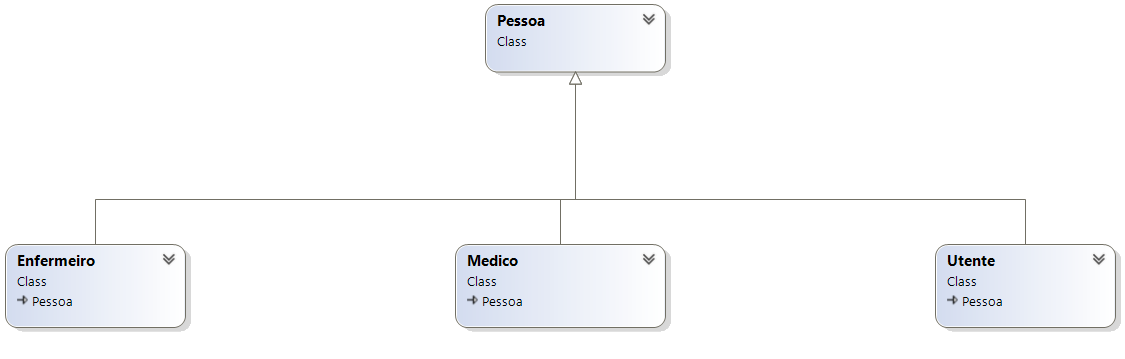


Figura 1-Herança de Classes

Foram também criadas as classes: Hospital, Triagem, ConsultaMedica e ProcessoClinico, a classe ProcessoClinico receba toda a informação registada na classe Triagem e ConsultaMedica. A classe Urgência foi removida.

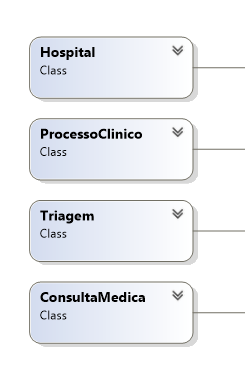


Figura 2- Classes Hospital, Triagem, ProcessoClinico e ConsultaMedica

Separamos o projeto em 4 camadas(NTier): Camada Front end (ainda por iniciar), Camada Data management onde vamos ter por exemplo listas com dados dos objects, e onde vamos gerir os dados que estas contém, Camada Control, onde definimos regras de como os dados podem ser acessados e utilizados pela front end, e Objects, onde temos os objetos, uma classes transversal ás outras(tal como definido nas referências das mesmas).

|  |
| --- |
|  |
| A Camada **Objects**, cada um contém o seu próprio estado, construtores, override ao ToString() e propriedades. |

|  |
| --- |
|  |
| A camada **Data Management**, contendo as listas de utentes, médicos, enfermeiros, listas de espera e os métodos necessários para as operar. Além disto temos também custom exceptions, dependendo do tipo de lista que poderá estar em falta ou a não funcionar corretamente.   |  | | --- | |  | | Camada **Control**, que controla como a camada Data Management é acessada. |  |  | | --- | |  | | Camada **Front End**, que contém o main e a janela de interação com o utilizador. | |

# Repositório

O trabalho e o relatório encontram-se no seguinte repositório GitHub:

<https://github.com/AndreFelicidade/18861_19544-_LP2>

# Problemas encontrados

Ao longo do desenvolvimento do trabalho, verificamos que até definirmos esta organização da resolução do problema, passamos por várias etapas de raciocínio e de estruturação de classes. Nesta segunda etapa, o problema envolveu reorganizar as classes por camadas, neste caso NTier, e corrigir discrepâncias em relação às normas CLS.