



INSTITUTO POLITÉCNICO
DO CÁVADO E DO AVE
ESCOLA SUPERIOR DE TECNOLOGIA

RELATÓRIO DE TRABALHO PRÁTICO

Gestão Frota de Veículos

JOSÉ LOUREIRO

ALUNO Nº 14821

Trabalho realizado sob a orientação de:
Luís Ferreira

Linguagens de Programação II

Licenciatura em Engenharia de Sistemas Informáticos

Barcelos, abril de 2020

Índice

1	MOTIVAÇÃO E PROPOSTA	1
2	ESTADO DA ARTE – C#	2
2.1	Programação Orientado a Objetos	2
2.2	Estruturação do Projeto em camadas	3
3	VISÃO GERAL DA APLICAÇÃO	4
3.1	Atores	5
3.2	Interface do Utilizador	6
4	CONCLUSÃO	7
	BIBLIOGRAFIA	9
	ANEXOS	11

Lista de Tabelas

Lista de Figuras

Figura 1: Logótipo da Escola de Tecnologias (Instituto Politécnico do Cávado e do Ave)1

Figura 2: Anders Hejlsberg.....2

Figura 3: Pilares da POO.....2

Figura 4:Camadas.....3

Figura 5: Gestão da frota de veículos.....4

Figura 6: Atores.....5

1 Motivação e Proposta

Pretende-se que sejam desenvolvidas soluções em C# para problemas reais de complexidade moderada. Serão identificadas as classes envolvidas, definidas as estruturas para suportar os dados e implementados os principais processos que permitam suportar essas soluções.

Pretende-se ainda contribuir para a boa redação de relatórios que descrevam o trabalho desenvolvido, a boa documentação de código fonte com a geração da API, e a gestão e planeamento de trabalho via ferramentas apropriadas (Git, GitHub, Trello ou outras).

O problema a explorar é do critério do aluno. Exemplos demonstradores de problemas que podem ser explorados:

- O mesmo problema explorado no trabalho realizado em LP1, agora implementado de acordo com o paradigma POO;
- Sistema que permita a gestão de ocorrências de incêndios florestais e outras catástrofes naturais;
- Sistema que permita gerir uma urgência de um hospital: listas de espera, prioridades, etc....;
- Sistema que permita gerir pessoas infetadas numa situação de crise de saúde pública. Neste sistema é importante: registar novos casos, contabilizar casos, consultar casos por região, sexo, idades, outros.

2 Estado da Arte – C#

C# Foi desenvolvida pela Microsoft por uma equipe liderada por Anders Hejlsberg (responsável também pelo desenvolvimento do Delphi e Turbo Pascal), que inicialmente a denominou como linguagem Cool. O desenvolvimento iniciou em 1999 e a linguagem foi lançada em 2002 como parte da plataforma .NET.



Figura 2- Anders Hejlsberg

2.1 Programação Orientado a Objetos

O C# dá suporte completo à programação orientada a objeto, incluindo encapsulamento, herança e polimorfismo.



Figura 3 – Pilares da POO

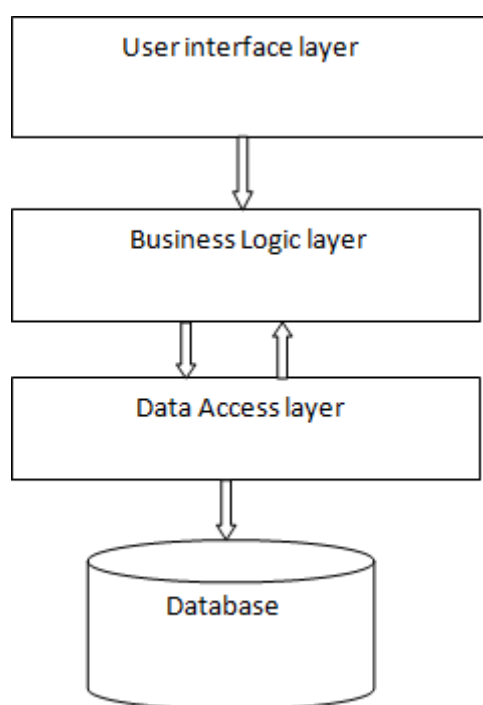
Encapsulamento significa que um grupo de propriedades, métodos e outros membros relacionados é tratado como uma única unidade ou objeto.

Herança descreve a capacidade de criar classes com base em uma classe existente.

Polimorfismo significa que você pode ter várias classes que podem ser usadas de forma intercambiável, ainda que cada classe implemente as mesmas propriedades ou métodos de maneiras diferentes.

2.2 Estruturação do Projeto em camadas

Um projeto separado em camadas tem como principal objetivo separar responsabilidades e tornar o código mais fácil de manter e ser reutilizado.



O projeto será dividido em três camadas, como podemos ver na figura do lado esquerdo:

Camada de Interface (UI)- responsável pela interação e apresentação de dados ao utilizador;

Camada de Negócios (BLL)- responsável em tratar as regras de negócio da aplicação;

Camada de Acesso aos dados (DAT)- responsável em acessar e persistir os dados da aplicação.

Assim, existe um nível de abstração entre as camadas. Isto torna possível alterar ou atualizar uma camada sem ter de interferir com a outra camada.

Figura 4 - Camadas

3 Visão Geral da Aplicação

Este trabalho tem como objetivo o desenvolvimento de um software direcionado para empresas de Táxis (ou outra qualquer que necessite de gerir a sua frota de veículos).

Este sistema será responsável por gerir, quer os veículos, quer os funcionários (motoristas e recursos humanos), assim como a "relação" entre eles (atribuição de veículos). Desta forma será possível visualizar de uma forma geral, qual o veículo que está atribuído a cada motorista. Nesta aplicação será possível:

- * Registrar/Consultar/Remover gerentes;
- * Registrar/Consultar/Remover colaboradores;
- * Registrar/Consultar/Remover veículos;
- * Reservar veículo (atribuir um veículo a um funcionário) / Listar reservas / Remover reservas;



Figura 5 – Gestão da frota de veículos

3.1 Atores

Administrador:

- * Poderá adicionar gerentes, listá-los e removê-los da atividade;

Gerente Operacional:

- * Responsável por aprovar a atribuição de veículos;
- * Poderá registrar veículos, consultá-los e removê-los;

Motorista:

- * Poderá solicitar a atribuição de um veículo;
- * Poderá consultar as reservas dos veículos;

Recursos Humanos:

- * Poderá adicionar colaboradores, listá-los e removê-los da atividade;

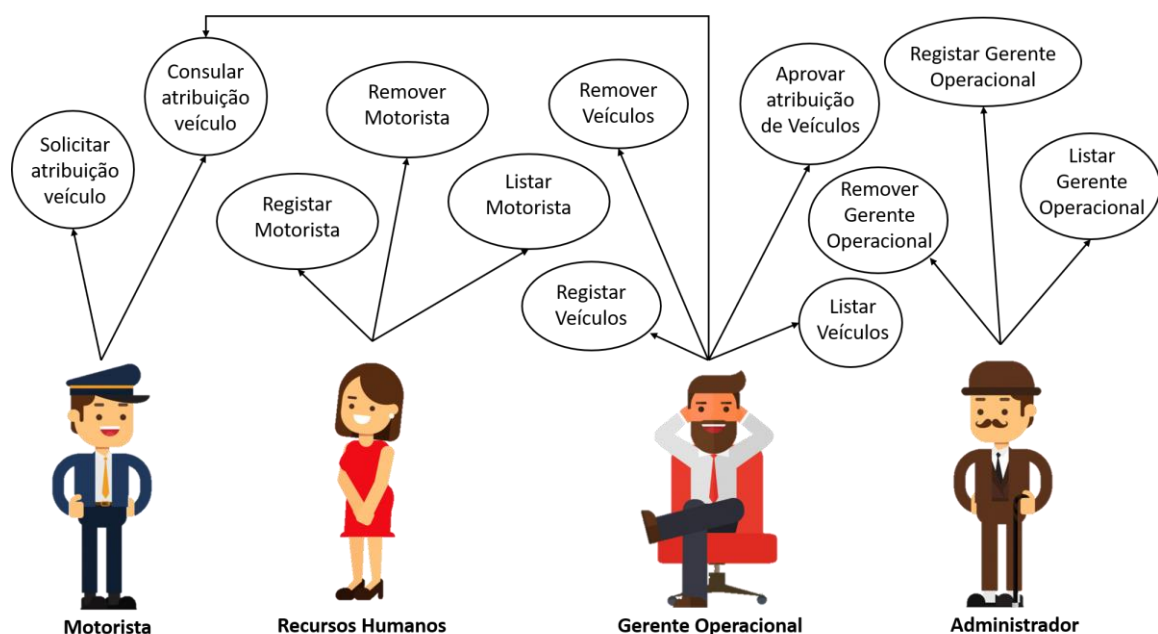


Figura 6 – Atores do sistema de Gestão da Frota de Veículos

3.2 Interface do Utilizador

1.Administrador (Login)

- 1.1. Registar Gerente
- 1.2. Listar Gerentes
- 1.3. Dispensar Gerente
- 1.4. Voltar

2. Gerente (Login)

- 2.1. Registar Veículo
- 2.2. Listar Veículo
- 2.3. Remover Veículo
- 2.4. Gerir Atribuição Veículos
 - 2.4.1. Listar Atribuição
 - 2.4.2. Atribuições Pendentes
 - 2.4.3. Voltar
- 2.5. Voltar

3. Recursos Humanos (Login)

- 3.1. Regista Motorista
- 3.2. Listar Motoristas
- 3.3. Dispensar Motorista
- 3.4. Voltar

4. Motorista (Login)

- 4.1. Solicitar Veículo
- 4.2. Consultar Atribuição Veículo
- 4.3. Voltar

5. Sair

3.3 Estrutura de Dados

Para guardar os dados, no meu ver, a opção mais vantajosa será a utilização de *Arrays* e *Dictionary*. Esta parte será desenvolvida na segunda fase da implementação do projeto.

4 Conclusão

Após terminar esta primeira fase do desenvolvimento deste trabalho prático, posso afirmar que consolidei muitos conceitos dados, até á data, na unidade curricular “Linguagem de Programação 2”. Até ao momento desenvolvi uma base para mais tarde continuar.

Bibliografia

Anexos