

*Upgrading Skills*

**Primavera**  
ACADEMY

# Projeto Money+ <sup>TM</sup>

RE\_START – Programação em C#

## Descrição do Sistema

Pretende desenvolver-se uma aplicação web que ajude a auscultar a saúde financeira de um indivíduo, ao permitir rastrear todas as suas despesas, investimentos, bens e dívidas, e com isso projetar a idade da reforma, tendo em atenção o seu estilo de vida.

O aspeto financeiro é um ponto basilar de qualquer pessoa, sendo extremamente importante auferir a situação em que nos encontramos para decidir o caminho correto a seguir. Tal como um GPS, não basta saber para onde queremos ir, pois o caminho é diferente, mesmo que o destino seja o mesmo.

A aplicação deverá permitir o registo de todas as despesas, corretamente categorizadas da seguinte forma:

- **Wallet** – Pode ser uma entidade bancária ou referenciar uma conta ou subconta bancária ou mesmo o dinheiro que tem em carteira;
- **Category** – Define uma categoria, tal como despesas pessoais, transporte, etc. Uma despesa deve estar sempre associada a uma sub-categoria;
- **Payee** – Representa o indivíduo ou entidade que providenciou o serviço ou bem que originou a despesa. Além do nome, deve ser possível registar o NIF da entidade.
- **Date** – Data em que a transação ocorreu;
- **Amount** – O valor total da transação;
- **Description** – Texto descritivo sobre a despesa.
- **Asset** – Atributo opcional que representa o bem a que a despesa está associada. Por exemplo, um carro, por exemplo, além de ser uma despesa quando se adquire, origina posteriormente muitas outras despesas tais como manutenção, combustível, etc.

Os requisitos aplicacionais para este módulo são os seguintes:

- ❖ R1 – Implementar operações de CRUD para todas as entidades encontradas;
- ❖ R2 – Obter listagem de despesas mensais detalhadas e ordenadas por data. A vista deverá ter links para o mês anterior, e o seguinte, de forma a tornar a navegação intuitiva. Na mesma vista, implementar um sistema de filtros por *Category*, *Payee* e *Asset*;
- ❖ R3 – Relatório agregado de despesas por categoria, subcategoria e mês num determinado ano. O relatório a apresentar deverá ser idêntico ao seguinte:

2022	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total
<b>Food</b>	<b>177.00</b>	<b>100.00</b>	<b>120.00</b>	<b>85.00</b>	<b>65.00</b>	<b>95.00</b>	<b>100.00</b>	<b>173.00</b>	<b>26.00</b>	<b>90.00</b>	<b>99.00</b>	<b>101.00</b>	<b>1,231.00</b>
Restaurants	125.00	0.00	0.00	0.00	0.00	25.00	0.00	52.00	0.00	0.00	0.00	0.00	202.00
Groceries	52.00	100.00	120.00	85.00	65.00	70.00	100.00	121.00	26.00	90.00	99.00	101.00	1,029.00
<b>Transportation</b>	<b>3.00</b>	<b>1.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>0.00</b>	<b>4.00</b>
Uber	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.00
Bus	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00
...													
<b>Total</b>	<b>180.00</b>	<b>101.00</b>	<b>120.00</b>	<b>85.00</b>	<b>65.00</b>	<b>95.00</b>	<b>100.00</b>	<b>173.00</b>	<b>26.00</b>	<b>90.00</b>	<b>99.00</b>	<b>101.00</b>	<b>1,235.00</b>

- ❖ R4 - Relatório agregado de despesas agrupadas pela categoria principal e ano.

	2020	2021	2022
<b>Food</b>	<b>950.00</b>	<b>1,050.00</b>	<b>1,231.00</b>
<b>Transportation</b>	<b>500.00</b>	<b>755.00</b>	<b>1,050.00</b>
...			
<b>Total</b>	<b>1,450.00</b>	<b>1,805.00</b>	<b>2,281.00</b>

### Módulo de investimentos:

Além das despesas, é necessário contabilizar os bens atuais e as entradas, também eles associados a uma Wallet.

Por exemplo, salário, depósitos a prazo, dinheiro em carteira, entradas de dividendos, e mais-valias de ações.

- ❖ R5 - Implemente um sistema que permita transferir dinheiro de uma Wallet para outra. Por exemplo, quando compra 1000€ de ações da EDP, está a retirar esse montante de uma conta à ordem e a transferi-lo para esse investimento.
- ❖ R6 - Relatório que permita analisar o montante atual por Wallet e calcular o n.º de anos que o utilizador pode estar sem trabalhar, considerando o seu custo de vida. O cálculo deve considerar uma taxa de inflação e de retorno de investimento parametrizável;

### Módulo de categorias de despesas e investimentos.

Só os administradores é que podem criar, editar ou apagar categorias de despesas e investimentos.

- ❖ R7 - Criar uma vista /admin/secret que permitirá criar um administrador com o respetivo role de Admin. Esta vista deverá ser protegida com a password "pouco-segura" que for

passada na query string. Exemplo: /admin/secret/?pwd=pouco-segura;

- ❖ R8 - Criar as vistas para que o admin possa efetuar operações de CRUD sobre as categorias;
- ❖ R9 - Importação de uma lista de categorias/subcategorias em formato yaml automaticamente, de 5 em 5 minutos. Não se pode inserir categorias repetidas e o ficheiro deve ser lido da pasta c:\temp\categorias;

## Snapshot

## diário

- ❖ R10 – Enviar um email diário a cada um dos utilizadores com o valor atual do seu património.

Nota: em vez de enviar um email, poderá simular usando uma tabela onde regista o corpo do email, a data de envio e o utilizador;

## Exercício

Faça a implementação do sistema proposto em ASP.NET 7, usando a Entity Framework como ORM para o acesso e atualização dos dados.

## Dicas e sugestões para a resolução

- Desenvolva o projeto usando código limpo, bem indentado e fazendo amplo uso do princípio de responsabilidade única;
- Use comentários quando for estritamente necessário, privilegiando a escrita de código claro e sem ambiguidade;
- Use o conceito de repositórios (*Repository/Service Pattern*) para organizar melhor o seu código;
- Use o mecanismo de exceções quando tal se justifique;
- Crie um repositório no GitHub para registrar todas as alterações que for implementando ao longo do desenvolvimento do projeto;
- Use um mecanismo de logger para registrar exceções e outras informações críticas sobre o sistema em produção.
- Para detalhes omissos de implementação ou que não estejam claramente definidos, use o seu bom-senso para resolver o problema e explique as suas opções;

## Critérios de classificação

O trabalho desenvolvido será classificado de acordo com os seguintes pesos:

Critério	Peso
Representação do modelo e <i>DB context</i>	15%
Implementação de funcionalidades	70%
Uso do git com o histórico de todos os <i>commits</i> realizados desde o início do projeto	5%
Código limpo e readme file	10%
Funcionalidades extra (+10%)	10%
<i>Total:</i>	110%

## Como entregar o projeto

A entrega do projeto deverá ser feita até ao final do dia 11 de dezembro de 2022. Para esse efeito, deverá criar um ficheiro ZIP que inclua o projeto completo, com o seguinte nome:

***MoneyPlus\_PrimeiroNome.UltimoNome.zip***

Este ficheiro deverá ser enviado para [sergiomaria@gmail.com](mailto:sergiomaria@gmail.com) com o assunto do email igual ao nome do ficheiro.

**Nota:** Deverá ainda partilhar o repositório do projeto com o utilizador [sergiomaria@gmail.com](mailto:sergiomaria@gmail.com)

**Bom Trabalho!**