



INSTRUÇÃO PRÁTICA		.NET-P004
MÓDULO	.NB – Formação .Net - Básico	
OBJETIVO DA ATIVIDADE		TEMPO
Colocar em prática os conceitos de Orientação a Objetos através do desenvolvimento, em grupo, de um protótipo de aplicação comercial. Além disso, estimular o desenvolvimento em equipe utilizando Git e Github.		4H
DESCRIÇÃO		
<p>Atividade em Grupo</p> <p>Crie um repositório do grupo para esta atividade.</p> <ul style="list-style-type: none">• No repositório do grupo, crie uma Branch para cada membro.• O projeto final deve estar na Branch principal (main ou master) após merge com as branches individuais.• A entrega deve ser feita, individualmente, no Moodle, informando o endereço do repositório e a BRANCH em que desenvolveu suas atividades.• Utiliza o fórum do Moodle para tirar suas dúvidas sobre as questões; <p>Objetivo: Desenvolver um sistema de gerenciamento da academia Tech Med.</p> <p>Desenvolvimento Implementar, obrigatoriamente, o gerenciamento das entidades abaixo e seus atributos, com as opções (telas) de inserir e remover para cada entidade. Além disso, a geração dos relatórios solicitados e a regra de negócio de acordo com o especificado abaixo. Classes acessórias devem ser implementadas a critério do grupo. Exemplo: uma classe base Pessoa para implementar herança e polimorfismo entre Médico e Paciente; uma classe Consultório (ou App) que vai implementar as telas, lógica e regras de negócio. O grupo não deve se restringir a, apenas, estas classes.</p> <p>Classes</p> <ul style="list-style-type: none">• Médico:<ul style="list-style-type: none">○ Nome (string)○ Data de Nascimento (DateTime)○ CPF (tratar entrada com 11 dígitos) (string)○ CRM (string)• Paciente:<ul style="list-style-type: none">○ Nome (string)○ Data de Nascimento (DateTime)○ CPF (tratar entrada com 11 dígitos) (string)○ Sexo (string)○ Sintomas (List<string>)		



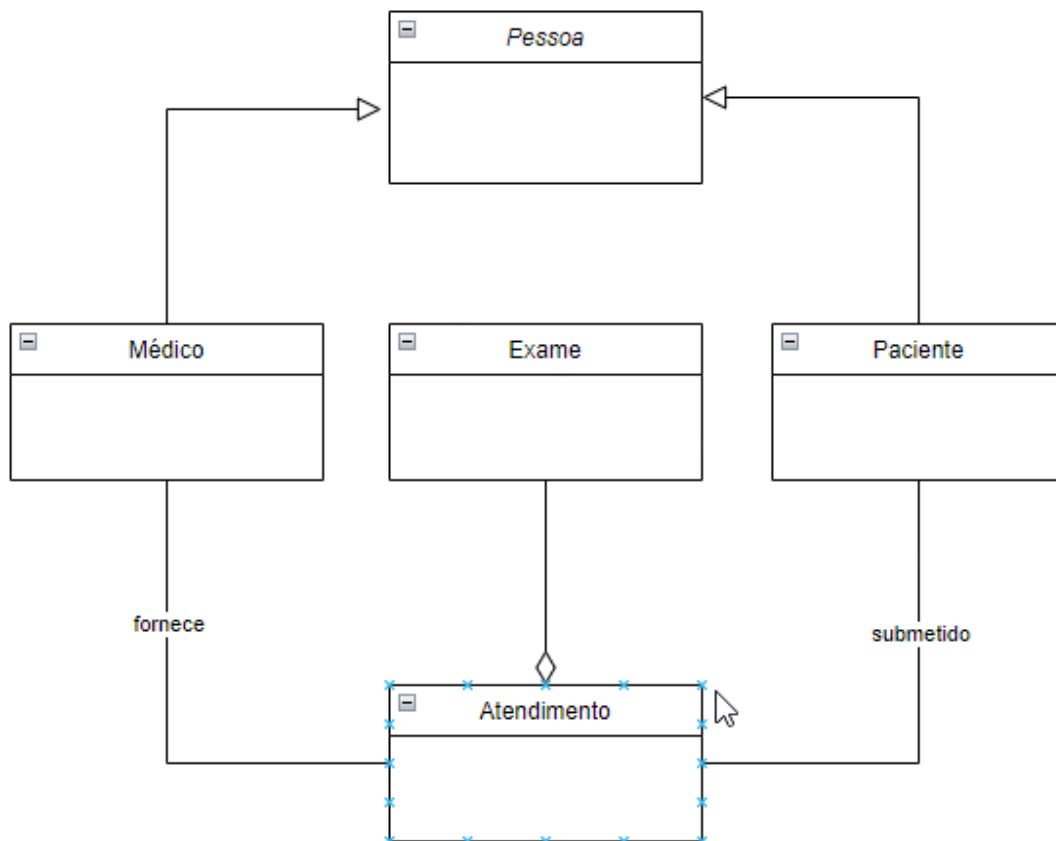
- **Exame**

- Título (string)
- Valor (float)
- Descrição (string)
- Local (string)

- **Atendimento**

- Início – Dia e hora (DateTime)
- Suspeita Inicial (string)
- Lista de Exames/Resultado (List<(Exame,String)>)
- Valor (float)
- Fim – Dia e hora (DateTime)
- Médico Responsável (Médico)
- Paciente (Paciente)
- Diagnóstico final (string)

UML



Regras de negócio:

- Deve haver um método para iniciar um Atendimento o outro método para finalizar o atendimento.
- Ao iniciar o atendimento deve ser cadastrada uma suspeita, ao finalizar deve fechar um diagnóstico.
- A data final de um Atendimento tem que ser posterior à data inicial;
- Um médico pode fornecer diversos atendimentos para diversos pacientes;
- Um atendimento só pode ser fornecido por um médico a um paciente.



Relatórios:

1. Médicos com idade entre dois valores.
2. Pacientes com idade entre dois valores.
3. Pacientes do sexo informado pelo usuário.
4. Pacientes em ordem alfabética.
5. Pacientes cujos sintomas contenha texto informado pelo usuário.
6. Médicos e Pacientes aniversariantes do mês informado.
7. Atendimentos em aberto (sem finalizar) em ordem decrescente pela data de início.
8. Médicos em ordem decrescente da quantidade de atendimentos concluídos.
9. Atendimentos cuja suspeita ou diagnóstico final contenham determinada palavra.
10. Os 10 exames mais utilizados nos atendimentos.

Dicas:

- Utilize expressões lambda para cálculos específicos e consultas às listas.
- Implemente consultas LINQ para gerar relatórios personalizados.
- Utilize tratamento de exceções para garantir uma experiência de usuário mais robusta e amigável.

REFERÊNCIAS

- <https://learn.microsoft.com/pt-br/dotnet/csharp/language-reference/built-in-types/value-tuples>
- <https://learn.microsoft.com/pt-br/dotnet/csharp/language-reference/operators/lambda-expressions>
- <https://learn.microsoft.com/pt-br/dotnet/csharp/programming-guide/concepts/linq/introduction-to-linq-queries>
- <https://learn.microsoft.com/pt-br/dotnet/csharp/fundamentals/exceptions/>