**ROTEIRO DE AULA PRÁTICA – CAPÍTULO 4**

**DISCIPLINA: ANÁLISE E PROJETO DE SISTEMAS**

**TÍTULO DA AULA: DESIGN DE SOFTWARE**

**1. Objetivos da Aula**

• Compreender os principais padrões de design de software (Factory, Singleton, Observer, Strategy, etc.).  
• Desenvolver exemplos práticos desses padrões nas linguagens Python ou C.  
• Explorar conceitos de arquitetura multicamadas.  
• Criar diagramas UML que representem a aplicação dos padrões em um sistema exemplo.

**2. Recursos Necessários**

• Computadores com ambiente configurado para Python e/ou C.  
• Ferramentas de modelagem (Draw.io, Astah ou Lucidchart).  
• Editor de código (VS Code, Visual Studio, Code::Blocks ou similar).  
• Material de apoio: Capítulo 4 do livro-texto.

**3. Estrutura da Aula**

1. Abertura (10 minutos): Revisão dos conceitos de padrões de projeto.  
2. Revisão Conceitual (20 minutos): Apresentação de exemplos práticos (Factory, Singleton, Observer).  
3. Demonstração (20 minutos): Criação de um pequeno projeto aplicando um padrão.  
4. Atividade Prática (40 minutos): Implementar um padrão em Python ou C e criar o diagrama UML correspondente.  
5. Encerramento (20 minutos): Apresentação das soluções e discussão em grupo.

**4. Relatório Final**

O relatório deve conter:  
• Resumo teórico dos padrões abordados.  
• Código-fonte desenvolvido.  
• Diagrama UML representando o padrão aplicado.  
• Reflexões sobre o uso dos padrões no desenvolvimento.

**5. Critérios de Avaliação**

• Clareza do resumo teórico (2,0).  
• Qualidade e funcionalidade do código desenvolvido (3,0).  
• Correção e clareza do diagrama UML (3,0).  
• Organização e apresentação do relatório (2,0).

**6. Conclusão**

Ao final desta prática, o estudante será capaz de identificar e aplicar padrões de design em sistemas reais, bem como representar essas soluções em diagramas UML.