



## Atividade Prática referente a aula 1

Aluno:	Miguel Alexandre Uhlein	Data:	06/08/2024
--------	-------------------------	-------	------------

Responda a atividade a seguir **com as suas palavras**, expressando seu entendimento do assunto estudado.

Atenção mínimo 3 (três linhas) por resposta.

**OBS.: Entregar via T2K correção até o final da aula.**

**AULA 01 – Revisão de conteúdo.**



- 1) Ferramentas são apoios importantes para o desenvolvimento de software. Por que a camada de ferramentas da **Figura 1.1** não funciona adequadamente sem as camadas inferiores?  
É necessário estruturar o plano antes de escolher as ferramentas, entender as necessidades do cliente, elaborar um projeto detalhado do software, sempre priorizando a qualidade e atendendo às expectativas. Só depois disso, deve-se selecionar as ferramentas e iniciar o desenvolvimento, realizando testes e manutenção após a entrega.
  
- 2) A Engenharia de Software é apoiada em diversas tecnologias. Porém, exige-se muita criatividade também para desenvolver um software. A Engenharia de Software pode inibir essa criatividade?  
É mais fácil afirmar que a engenharia pode estimular sua criatividade, pois quanto mais liberdade e funcionalidades as tecnologias oferecem, maior é o leque de opções, ideias e possibilidades disponíveis. Isso impulsiona sua criatividade para criar soluções personalizadas e atender às demandas do cliente de forma inovadora.
  
- 3) De que maneira a Engenharia de Software pode ser efetiva nos projetos de software?  
A Engenharia de Software assegura a efetividade dos projetos ao definir claramente os requisitos, planejar e estimar prazos e orçamentos, e implementar práticas rigorosas de controle de qualidade e gerenciamento de riscos. Isso resulta em software de alta qualidade, entregue dentro do prazo e orçamento estabelecidos.



- 4) Após 40 anos de Engenharia de Software, pode-se dizer que ainda se vive uma crise do software, considerando a falta de atendimento de prazos, custos e de qualidade?  
Sim, após 40 anos, ainda há uma "crise do software" devido a desafios contínuos com prazos, custos e qualidade, apesar dos avanços nas metodologias e ferramentas. Problemas como requisitos mal definidos e complexidade crescente ainda afetam muitos projetos.