

# **ENGENHARIA DE SOFTWARE I** Análise e Desenvolvimento de Sistemas *Prof. Evandro Zatti, M. Eng.*

## **ATIVIDADE PRÁTICA**

### Fundamentos da Engenharia de Software

Equipe: Alex Rogério, Bruna Jaqueline, Carlos Eduardo, Fellipe Broeto, Fausto Henrique, Glauco Custado, Matheus Souza, Vitor Cordeiro, Leonardo Artigas.

1. Explique com suas próprias palavras o que quer dizer a frase "Software é desenvolvido ou engenhado e não manufaturado no sentido clássico".

Assim como o hardware o software também é um produto, ou seja, requer um trabalho em equipe formado por pessoas, porém sua criação é totalmente diferente. Sendo assim, ele é desenvolvido e não manufaturado.

2. O que você entende por **processo de desenvolvimento de software** e qual a sua importância para a qualidade dos produtos de software?

O processo de desenvolvimento de software, pode ser descrito como a base de toda a elaboração do software, uma série de passos a serem respeitados e realizados. Se caso algum passo for ignorado, seu projeto terá grandes chances de ser comprometido em meio ao processo. Em cada passo identificamos o que nosso usuário deseja, o que é possível ser implantado, o tempo que será necessário para conclusão e quanto esse software irá custar para ser desenvolvido.

3. Qual a diferença entre **processo** e **projeto** de software?

Processos são ações coordenadas, utilizadas para organização e realização de um projeto em específico. Para cada projeto, é necessário de um processo, mesmo que seja a

mesma equipe. Afinal cada cliente, ou cada projeto busca um resultado em particular.

4. O que é **qualidade de software**?

Para que um software possa ser visto como um “software de qualidade”, requer um ponto de vista específico. Para isso, podemos afirmar que um software de qualidade depende das necessidades impostas pelo usuário, para que este software consiga atendê-las, levando em consideração se o processo possui uma melhoria contínua, ou seja, se este processo após ser entregue ao usuário tenha o mínimo possível de erros.

5. O que são ferramentas **CASE**? Cite 2 exemplos de ferramentas CASE.

Case é o termo usado para nos referirmos a uma escolha de um comando para cada caso específico, podemos afirmar que é simplesmente uma escolha de uma alternativa pelo usuário. (ex: ER/ Studio)

6. O que é um **protótipo**? Que cuidados são importantes e próprios desse modelo no que diz respeito a comunicação com os clientes?

Protótipo é um modelo/molde inicial/final de um produto, onde visa testar e avaliar aspectos como viabilidade, requisitos funcionais, arquiteturas, entre outros, sem a necessidade de implementação/codificação.

Uma das vantagens de se utilizar de um protótipo na criação de determinado produto é a taxa de feedback que se pode obter através do cliente, facilitando nas mudanças e aprimoramentos das funções que o sistema irá executar.

Entretanto, cuidados devem ser tomados e limites estabelecidos em relação da comunicação com o cliente sobre o produto, pois uma filtragem final deverá ser feita para modificar o que de fato aumentará o desempenho ou deixar mais intuitivo no projeto final.

7. Procure na internet pelo menos duas definições para o termo **Engenharia de Software**.

1ª Definição - Para Bauer (1969) *apud* Pressman (2006), a Engenharia de Software é "a criação e a utilização de sólidos princípios de engenharia a fim de obter software econômicos que sejam confiáveis e que trabalham eficientemente em máquinas reais".

2ª Definição - Engenharia de software é uma área do conhecimento da informática voltada para a especificação, desenvolvimento e manutenção de sistemas de software aplicando tecnologias e práticas de ciência da computação, gerência de projetos e outras disciplinas, objetivando organização, produtividade e qualidade.

8. Os mitos da área de software estão pouco a pouco enfraquecendo. No entanto outros estão tomando seu lugar. Encontre pelo menos um mito novo em qualquer uma das categorias (administrativos, clientes ou profissionais).

#### MITOS DE GERENCIAMENTO

- Já temos um livro que está cheio de padrões e procedimento para desenvolver software; ele não supre meu pessoal com tudo o que eles precisam saber?  
Não bata ter um livro com padrões. O principal recurso para se desenvolver software são pessoas e eles precisam além de conhecer os padrões precisam saber utilizá-los no momento certo.
- Se o cronograma atrasar, poderemos acrescentar mais programadores e ficarmos em dia.  
Programadores possuem tempo para se adaptar e harmonizar com uma nova equipe. Não se pode pensar que ele entrará no projeto em alta produtividade a ponto de salvar o atraso.
- Se eu decidir terceirizar o projeto de software, posso simplesmente relaxar e deixar essa empresa realizá-lo.  
O desenvolvimento de software é uma atividade que requer um acompanhamento contínuo para garantia da qualidade, já que não é algo palpável ou concreto.

## MITOS DOS CLIENTE

- Uma definição geral dos objetivos é suficiente para começar a escrever os programas – podemos preencher detalhes posteriormente.

“Dá uma olhadinha aqui e pode já sair fazendo.” Os clientes nem sempre tem muito prazer em ensinar o funcionamento de seu trabalho e isso pode acontecer. Porém, bom entendimento os requisitos garantem bom resultado final.

- Os requisitos de software mudam continuamente, mas as mudanças podem ser facilmente assimiladas, pois o software é flexível.

Algumas demandas não são facilmente adicionadas ao produto, muitas das vezes requerem um grande desprendimento de esforços.

## MITOS DO PROFISSIONAIS DA ÁREA DE SOFTWARE

- Uma vez feito um programa e colocado em uso, nosso trabalho está terminado.

Maior parte do ciclo de vida de um software é sua manutenção.

- Até que o programa entre em funcionamento, não há maneira de avaliar sua qualidade.

Há testes feitos ao longo da construção do software e não apenas no final.

- O único produto passível de entrega é o programa em funcionamento. Protótipos podem ser utilizados durante iterações

- A engenharia de software nos fará criar documentação volumosa e desnecessária, que inevitavelmente irá nos retardar.

Engenharia de Software não é sinônimo de documentação. A documentação é consequência de um processo de explicitação da solução. À medida que se “desenha” a solução em um documento, novas ideias surgem, erros são encontrados e qualidade é em parte garantida.