1. Explique o contexto em que a Engenharia de Software surgiu. Fundamente sua resposta com base nos acontecimentos históricos da época.

R:

O contexto que a engenharia de software surgiu foi no ambiente em que todos estavam passando por um momento turbulento e precisavam de uma solução para os problemas, estavam na época dos anos 60, se tendo várias mudanças no cenário do desempenho dos computadores e com isso veio um rápido crescimento da demanda do software, se tinha complexidade maior com os problemas que deviam ser resolvidos e a inexistência de técnicas que funcionam.

Como consequência desses dias surgiram projetos estourando o orçamento, projetos entregues fora do prazo, software de baixa qualidade e por aí vai. Desse modo, ninguém via as aplicações e mudanças que o software tem hoje em dia, foi que ai na OTAN surgiu uma reunião falando sobre os problemas e discutindo a ideia da engenharia de software.

2. Defina o que é um software, sua classificação e tipos.

R:

* Importante destacar que programa não é a tradução de software, software para nós significa programa junto dos seus artefatos, seu eu tiver somente um código fonte isso por si só não é um software, posso ter um manual artefato, documentação de pré-requisitos

Um software nada mais é do que um programa e seus artefatos, um software novo pode ser criado através de desenvolvimento de novos programas, configuração de sistemas de software genéricos, reutilização de um novo software existente.  
  
Pode ser classificados em genéricos uso geral (microsoft word) e personalizado(software de uma loja de roupas), lembrando que um genérico foi pensado não somente em uma pessoa ele atende várias e um software personalizado ele teve o pensamento para aquele sentido, mas ambos podem se transformar e se tornar outra classificação  
  
TIpos:

Aplicativos  
SIstemas  
Sistemas embarcados  
Sistemas móveis  
Driver  
Sistemas operacionais

3. Defina a Engenharia de Software e suas áreas de conhecimento.

R:

A engenharia de software de software se dedicas às teorias, métodos e ferramentas para desenvolvimento de software profissional(Sistemas não triviais e com base em um conjunto de requisitos), respeitando o cronograma acordado, satisfazendo a necessidade dos clientes e minimizando o custo da manutenção e dedicando-se ao desenvolvimento de software a custos adequados.

Ela é além disso uma disciplina que relaciona com todos os aspectos da produção de software, propoẽ ferramentas, técnicas e processos para, entender com precisão qual é o problema, produzir uma solução adequada e levando-se em consideração as restrições de desenvolvimento e recursos disponíveis.   
  
1. Engenharia de requisitos  
2. Projeto  
3. Construção

4. Testes

5. Manutenção

6 Gerência de configuração

7. Gerência de projetos

8. Processos de software

9. Modelos de processo

10. Qualidade

11. Prática Profissional

12. Aspectos Econômicos

4. Explique e defina pelo menos 4 princípios que regem a Engenharia de Software.

1. A razão de existir

* O software precisa ter uma razão para existir, ou seja, seu software precisa resolver o problema de alguém ou de alguma coisa

1. Kiss (Keep it Simple, stupid!)

* Quanto mais simples melhor, quanto mais simples for o software, mais fácil de ele ser testado, coloque e desenvolva e pensa em técnicas e ferramentas que sejam simples, simples de usar, simples de instalar e de entender, mantenha o simples que às vezes pode ser complexo.

1. Mantenha a visão

* Mantenha a visão sobre o seu produto, ou seja, daquilo que está sendo produzido, quem é meu público alvo, o que é preciso entregar, para não chegar com produto que não foi requisitado ou está fora do prazo.

1. O que um produz, outros consomem

* Em um processo de desenvolvimento de software, nós fazemos isso para as pessoas, pense nos usuários, pense onde esse sistema vai ser incluído e pense na acessibilidade, seu sistema deve ter isso para que seja produzido e consumido.

5. Quais os maiores desafios enfrentados pelos Engenheiros de Software atualmente? Contemple na sua resposta questões éticas, profissionais, dilemas éticos.

1. Heterogeneidade

* Sistemas de software devem ser capazes de lidar com diferentes plataformas de hardware e ambientes de execução.
* O mesmo sistema deve funcionar para diferentes tipos de usuários, em contextos específicos.

1. Entrega

* O sistema deve ser entregue ao cliente no menor tempo possível, com o menor custo possível.

1. Confiança

* O usuário deve poder justificadamente depositar sua confiança no sistema.

1. Escala

* O sistema deve funcionar adequadamente mesmo quando um grande número de usuários o está usando.

Responsabilidade ética:

* A engenharia de software envolve responsabilidades mais amplas do que simplesmente a aplicação de habilidades técnicas.
* Os engenheiros de software devem se comportar de modo honesto e eticamente responsável para serem respeitados como profissionais.
* O comportamento ético é mais do que simplesmente a sustentação de leis.
* Confidenciabilidade:
  + Os engenheiros de software devem normalmente respeitar a confidencialidade de seus funcionários ou clientes, independentemente de ter ou não assinado um acordo formal.

* Competência:
  + Os engenheiros não devem conscientemente aceitar um trabalho que esteja fora de sua competência.

Responsabilidade profissional

* Desenvolvedores devem estar cientes das leis locais que regem o uso da propriedade intelectual, tais como patentes, direitos autorais, etc
* Eles devem tomar cuidado para assegurar que a propriedade intelectual dos funcionários e clientes seja protegida

DIlemas Éticos

* Discordância, em princípio, das políticas de gerência sênior, Como lidar?
* Um funcionário age de forma não ética e libera um sistema de segurança crítica sem finalizar o teste do sistema. O que fazer?
* Participação no desenvolvimento de sistemas de armazenamento militar ou sistemas nucleares. Você toparia?
* LGPD lei geral da proteção de dados

6. Dos mitos discutidos em aula, escolha pelo menos 2 relacionados à gerência, ao cliente e ao profissional do software.

Gerência:

Já temos um livro que está cheio de padrões e procedimentos para elaborar o software.

* Mesmo com todos os livros e técnicas todas engenharia de software não te dá a receita de bolo completa

Se nós atrasarmos o cronograma?

* Estudos demonstram que dependendo do projeto e da classificação da categorização dessa nova pessoa pode atrasar ainda mais o cronograma do projeto, querer adicionar programadores novos para adiantar o projeto às vezes pode ser bem ruim

Cliente:

O estabelecimento geral de objetivos é suficiente para iniciar a escrita de programas?

* Podemos fornecer os detalhes posteriormente.

Os requisitos de projeto mudam continuamente?

* As mudanças podem ser facilmente acomodadas porque o software é flexível.