**Estudo dirigido**

1) Em sequência, quais são as principais atividades do processo de software e a principal atividade de cada uma delas?

Especificação de software – clientes e engenheiros definem o software que deve ser produzido com suas restrições

Projeto de software – Atividade que envolve a modelagem com UML e programação

Implementação de software – desenvolver o software em si utilizando padrões de projeto

Validação de software – Verificar se o que foi desenvolvido é o que foi pedido pelos clientes e se não contém erros (lógicos x implementação)

Evolução de software – devido a desejo de novas funcionalidades por parte do cliente ou exigências e mudança de mercado

2) Existem três principais modelos de processo: plano (cascata), iterativo e componentes. Cite dois aspectos que norteiam a escolha do processo a ser utilizado

Capacidade do time e característica do problema

3) Cite uma vantagem e desvantagem dos métodos em iterativos e plano.

Iterativos – vantagem – rapidamento o cliente vai vendo o resultado do projeto, pois possui módulos prontos e demostráveis. Custo de acomodação de modificações do cliente são reduzidos.

Desvantagem – Não possui uma boa documentação

cascata – vantagem – processo é utilizado em projeto de grandes sistemas quando o mesmo é desenvolvido de forma distribuída, isso ajuda na coordenação do projeto. Desdo o começo já se sabe o que o sistema terá. Isso traz confiabilidade.

desvantagem – uma fase tem que ser totalmente finalizada antes da outra, não aproveitando o paralelismo da equipe além de ser difícil de responder à modificações dos requisitos dos clientes

Integração (componentes) – vantagens – baseado em reuso de software. Sistemas são integrados a partir de componentes já existentes. Componentes reusados podem ser adaptados ou configurados para atender a funcionalidade desejada. Custo e risco reduzido visto que grande parte do software não é desenvolvido desde o início.

Desvantagem – Perda de controle sobre a evolução dos elementos reusados.

4) Cite vantagens e desvantagens do modelo baseado em reuso (integração e configuração)

Já respondida anteriormente. Respondi a anterior de forma mais completa para abordar outros tópicos.

5) Explique as três principais fases da engenharia de requisitos

Elicitação – Saber o que os clientes esperam que o sistema faça. Extração dos requisitos.

Especificação dos requisitos (Documentação) – Definição formal dos requisitos de forma detalhada

Validação – Verificar se não há contradições ou impossibilidades. Objetiva mostrar se o sistema está em conformidade com sua especificação.

6) Explique uma situação em que vc utilizaria protótipo?

Para mostrar ao cliente se o sistema está o sistema está de acordo com os requisitos, mostrar os principais aspectos, antecipando as modificações. Isso pode fazer com que o cliente pense de uma outra maneira.

7) Por que não se deve utilizar um protótipo na continuidade do desenvolvimento?

Porque ele é bem básico, servindo portanto, apenas para demonstrações mais simples. Tendo uma estrutura cheia de “gambiarras” devido ao rápido desenvolvimento.

8) Explique o processo de qualificação de processo CMMI utilizando os seus 5 níveis.

Serve para qualificar em que estágio o processo está. O processo pode está no nível 1 essencialmente não controlado, nível 2 onde os processos dependem dos envolvidos e cada projeto pode ter um processo diferente, nível 3 projeto definido e institucionalizado, nível 4 processo possui medidas bem definidas e nível 5, onde se estratégias de melhoria bem definidas e usadas.

9) Cite duas prática da técnica XP, explicando a importância das mesmas para a metodologia.

Programação em par, é importante porque se um membro sair da equipe, os outros saberão realizar o trabalho visto que todos participam de cada parte do projeto em par, onde esse par muda.

Não há documentação, devido a rapidez de entrega do projeto não há perda de tempo com documentação

10)O que é refactoring e por que este conceito é tão importante para o XP?

Tem por objetivo antecipar modificações para diminuir custos futuros. Por exemplo, através da reorganização da hierarquia de classes removendo código duplicado, renomeação de métodos e atributos tornando-os mais intuitivos, remoção de trechos de códigos por chamada de funções.

É importante pois pode melhorar o código aos poucos já que o código fica muito “sujo” à medida que vai aumentando e tendo em vista o rápido desenvolvimento do mesmo devido ao pouco tempo de entrega.

11) O que significa a prática “Ritmo Sustentável” em XP?

Significa que o sistema deve ser entregue sem necessidade de hora extra por parte dos membros da equipe, pois teria algo errado se isso acontecesse. O XP deve ser rápido mas praticável e possível dentro das horas regulares de trabalho.

12) Cite três reuniões que acontecem no SCRUM e o que acontece em cada uma delas.

Reunião de entrega – planejamento do lançamento

Planejamento do sprint – reunião antes do início do sprint com o objetivo de determinar o plano e definir as metas.

Retrospectiva – após revisão e antes do próximo planejamento. Reunião de melhoria para encontrar erros e armadilhas passados buscando evitar tais erros

Daily – Reunião a cada sprint com duração de aproximadamente 15 minutos. Basicamente encontro no mesmo horário e lugar para que o time se mantenha atualizado um ao outro sem ter que escrever ou ler muito depois para saber como está o desenvolvimento do projeto.

Revisão do sprint – Encontro informal que o time de desenvolvimento o scrum master e o scrum proprietário e interessados participarão. O time dá uma demonstração do produto mostrando o que está terminado e o que não está com o intuito de mostrar a clientes e interessados o que eles estão realizando sobre o sprint e comparar com o comprometimento dado no início do sprint.

13) Quais foram as duas que faltaram?

Falei de todas na resposta acima

14) Quem é e o que faz o Product Owner?

É o representante do cliente na empresa mas tendo autoridade sobre outros membros do time, mesmo como o scrum master. Tem o objetivo de continuar olhando um produto mesmo depois de pronto por um período de tempo e responsável por atingir o sucesso de um produto, devendo interagir com os clientes e usuários e membros do time de desenvolvimento e interessados.

15) Qual a diferença dos requisitos funcionais para os não-funcionais? Dê exemplos.

Funcionais – foca no que o sistema deve fazer. Ex.: um usuário ser capaz de procurar a lista de agendamento para todas as clínicas, o sistema deve ser unicamente identificado por 8 dígitos.

Não funcionais – foca no como fazer. Ex.: O sistema deve ser operável por usuários inexperieentes, em português, com design responsivo.

16) O que significa dizer que um requisito funcional é completo e consistente?

Completos – deve incluir a descrição de todos os fluxos possíveis

Consistentes – não deve contradizer ou conflitar as descrições de seus fluxos.

17) Qual é o mais utilizado método na elicitação (cenário, etnografia, entrevista, protótipo) de requisitos e qual o principal problema deste método?

A entrevista. Há problema quando se vai falar com pessoas de uma área específica com o vocabulário específico daquela área, como por exemplo, médico e juristas.

Ver etnografia e cenário

18) Que método vc utilizaria para fazer a especificação de requisitos?

Tabela, casos de uso (explicar)

19) Qual o principal papel do gerenciamento de requisitos?

Saber quais módulos vão ser afetados quando um requisito for alterado.

20) Fora o diagrama de casos de uso, cite dois outros diagramas UML e diga para que eles servem.

Diagrama de classes (citar outro e dizer para que servem)

21) Qual o principal objetivo da engenharia orientada a modelos?

Desenvolver código a partir de modelo

22) Cite um padrão de arquitetura e diga a vantagem e desvantagem do mesmo

camada – Bom para sistemas que requerem uma maior segurança devido às suas camadas. Desvantagem – ser lento

pipeline – simples camadas de passos não perdendo tanto desempenho mas não tão amplo para sistemas mais complexos.

(ver as outras arquiteturas)

23) Cite outro padrão de arquitetura e diga a vantagem e desvantagem do mesmo

MVC – apresenta múltiplas formas de visualização dos dados. Dados podem ser podem ser modificados independentemente.

Desvantagem – Não propícios para aplicações com modelos de dados e interações simples

24) Qual a diferença de testes de caixa branca e caixa preta?

Teste de caixa branca se sabe o que foi feito e como foi feito já caixa preta não. Caixa preta usa apenas inputs e outputs para verificação do resultado sem saber como o sistema ou parte que está sendo testada foi feita.

25) Qual a diferença de inspeção, ou análise estática, para testes que realizam análise dinâmica?

A inspeção é feita no documento. Não verifica os erros não funcionais. Não validam conformidade com o usuário. Em caso de erro pode continuar o projeto e depois corrigir tal erro. Teste dinâmico é feito em tempo de execução, logo não se pode avançar caso tenha erro, pois o mesmo pode se perpeturar ao longo do sistema, então corrige-se o erro e só depois avança.

26) Dentro do desenvolvimento, quais são os três principais tipos de testes e como eles se caracterizam?

Teste unitário – caracterizado por observar funcionalidades de objetos e métodos

Teste de componente – O foco é na interface de componentes (integram objetos)

Teste de sistema – teste de integração (sistema completo)

27) Qual a principal dificuldade para a realização dos testes?

Cada requerimento deve ter no mínimo um caso de teste. Porém cada requerimento pode ter n casos de teste ou um caso de teste com m configurações. Ou seja existem infinitos casos de testes, impossível testar todos.

28) Como o cálculo da complexidade ciclomática pode ajudar no processo de testes?

Ele indica a quantidade de caminhos que um sistema tem assim você sabe quantos caminhos testar.

29) Qual o princípio do desenvolvimento orientado a testes?

Testar primeiro antes de desenvolver, então tudo já vai estar testado. O código portanto deve ser escrito de maneira que passe no teste.

30) O que vc entende por teste de regressão?

Quando testar um módulo posterior, testar os anteriores também até o primeiro, mesmo que estes já tenham sido testados antes. Uso de automação é comum nesses testes.

31) Cite dois aspectos que devem ser observados durante os testes de usabilidade

Deixar o usuário à vontade para ele entender que é o sistema que está sendo testado e não ele;

Gravar ações dele e expressões para ter uma noção do que ele está achando do sistema que está testando;

Entrevista pós teste;

Ver o perfil do usuário para saber o grau de proximidade ele tem com o equipamento que possui o sistema que ele está testando ou o grau de proximidade com outros sistemas semelhantes.

32) Cite uma das utilidades dos pontos de função

Mensurar quanto aproximadamente um software deve custar, através de métrica mesmo que não exata, verificando também o tempo de desenvolvimento e complexidade do sistema.