ANDRÉ PARRA DOURADO
ENGENHARIA DE SOFTWARE III
1º/2022

EXERCÍCIOS

1. EXPLIQUE O CONCEITO DE PROCESSO DE SOFTWARE. QUAIS SÃO SUAS ATIVIDADES PRINCIPAIS? EXPLIQUE TODAS ELAS.

É um conjunto de atividades relacionadas que levam a produção de um software, podendo ser do zero ou não.

As **4** atividades principais são:

- ESPECIFICAÇÃO DE SOFTWARE: funcionalidade do software e suas restrições;
- PROJETO E IMPLEMENTAÇÃO DE SOFTWARE: desenvolvimento do software;
- VALIDAÇÃO DO SOFTWARE: validação para atendimento das demandas do cliente;
- EVOLUÇÃO DE SOFTWARE: constante evolução do software para atender mudanças dos clientes;

2. COMO PODEM SER CATEGORIZADOS OS PROCESSOS DE SOFTWARE?

- DIRIGIDOS A PLANOS;
- PROCESSOS ÁGEIS;

3. O QUE É UM MODELO DE PROCESSO? APRESENTE E EXPLIQUE DOIS MODELOS IMPORTANTES, APRESENTANDO SUAS VANTAGENS E DESVANTAGENS.

É uma representação simplificada de um processo de software, onde cada modelo representa uma perspectiva particular de um processo, fornecendo informações sobre ele.

MODELO EM CASCATA:

- Considera as atividades fundamentais (especificação, desenvolvimento, validação, evolução) e representa essas atividades como fases distintas.

MODELO EM CASCATA – BALANÇO:

- VANTAGEM: Modelo de gestão mais conhecido e fácil;
- VANTAGEM: Não necessita de funcionários muito autodisciplinados;
- VANTAGEM: Maior documentação;
- DESVANTAGEM: Tempo de implementação e feedback do cliente;
- DESVANTAGEM: Engessamento do projeto, dificuldade de alterações e adequações;

DESENVOLVIMENTO INCREMENTAL:

- É baseado na ideia de desenvolver uma implementação inicial, apresenta-la ao cliente e usuários e através do feedback continuar pelo meio da criação de várias versões até que um adequado seja desenvolvido.

- As atividades fundamentais (especificação, desenvolvimento, validação, evolução) são realizadas intercaladas e não separadas, com rápido feedback entre todas elas.

DESENVOLVIMENTO INCREMENTAL – BALANÇO:

- VANTAGEM: Custo de mudanças reduzido;
- VANTAGEM: Fácil obtenção de feedback;
- VANTAGEM: Entrega e implementação rápida de um software útil ao cliente;
- DESVANTAGEM: Menor documentação;
- DESVANTAGEM: Mais complexo para definir contratos com o cliente, já que o escopo inicial dificilmente será igual ao escopo do final do projeto;

4. EXPLIQUE O CONCEITO DE PROJETO DE SOFTWARE. EXPLIQUE O QUE DEVE SER FEITO AO PROJETAR UM SISTEMA.

Estágio do processo de software no qual um sistema executável é desenvolvido.

PASSOS QUE DEVEM SER FEITOS:

1. Compreender e definir o conceito e as interações com o sistema;
2. Projetar a arquitetura do sistema;
3. Identificar os principais objetos do sistema;
4. Desenvolver modelos de projeto;
5. Especificar interfaces;

5. EXPLIQUE O QUE É ARQUITETURA DE SOFTWARE.

É a estrutura que abrange os componentes do software, definindo ou retratando a forma como os componentes se relacionam.

É uma descrição de alto nível de abstração, permitindo uma visão completa do sistema.

6. QUAIS SÃO OS PRINCIPAIS MODELOS DE ARQUITETURA, EXPLIQUE-OS.

- MODELO ESTRUTURAL: Retratam a arquitetura como uma coleção de componentes de software;
- MODELO DINÂMICO: Buscam retratar os aspectos comportamentais do software, indicando o comportamento do software ao receber eventos externos;
- MODELO DE ARCABOUÇO: Denotam a busca por um projeto estrutural que possa ser reutilizado, ou seja, padrões, para diferentes aplicações similares;

7. EXPLIQUE OS CONCEITOS DE COMPONENTES, MÓDULOS E SUBSISTEMA. QUAIS SERIAM AS VANTAGENS DE SE UTILIZAR UMA ARQUITETURA COMPONENTIZADA.

- COMPONENTE: parte modular, possível de ser implantada e substituível de um sistema que encapsula implementação e exibe conjunto de interfaces;
- MÓDULO: conjuntos de componentes que utilizam e fornecem serviços de outros componentes;
- SUBSISTEMA: não devem depender de serviços fornecidos por outros subsistemas. Eles comunicam se entre si.

8. EXPLIQUE O CONCEITO DE GRANULARIDADE DE MÓDULOS, RELACIONANDO ESTE CONCEITO COM O PROJETO DE SISTEMAS. APRESENTE VANTAGENS E DESVANTAGENS.

Granularidade dos módulos está relacionado ao conceito de módulo e diz se algo é maior, menor ou se agrupa algo.

Quanto maior o número de módulos, maior o custo da integração e custo total de software.

Quanto menor o número de módulos, menor o esforço e menor custo.

9. QUAIS SÃO OS PRINCÍPIOS NECESSÁRIOS PARA ASSEGURAR A MODULARIDADE? EXPLIQUE.

– MAPEAMENTO DIRETO: a estrutura modular de um software deve ser compatível com o domínio do problema;

– POUCAS INTERFACES: cada módulo deve se comunicar com o mínimo possível de módulos;

– INTERFACES PEQUENAS E EXPLÍCITAS: se dois módulos se comunicam, eles devem trocar o mínimo possível de informações (acoplamento por dados);

– ESCONDER INFORMAÇÕES: os dados de um módulo só devem ser acessado a partir da interface;

10. EXPLIQUE OS CRITÉRIOS DE MODULARIDADE.

– DECOMPOSIÇÃO MODULAR: divisão de um problema maior em subproblemas com soluções individuais.

– COMPOSIÇÃO MODULAR: módulos podem ser combinados para produzir novos sistemas de software;

– FACILIDADE DE COMPREENSÃO DOS MÓDULOS: módulos podem ser entendidos em separado pelo leitor humano, sem ter que verificar outros.

– CONTINUIDADE MODULAR: o projeto satisfaz este critério se uma alteração na especificação do problema gera alteração em um só módulo

– PROTEÇÃO MODULAR: em condição anormal de execução, o problema fica confinado a este módulo ou no máximo em módulos vizinhos

11. EXPLIQUE O CONCEITO DE OCULTAMENTO DE INFORMAÇÃO, APRESENTANDO SUAS VANTAGENS EM RELAÇÃO AO PROJETO DE SOFTWARE.

Módulos devem ser especificados e projetados de tal forma que as informações (procedimentos e dados) contidas num módulo sejam inacessíveis a outros módulos que não tenham necessitam destas informações.

12. EXPLIQUE OS CONCEITOS DE INDEPENDÊNCIA FUNCIONAL, COESÃO E COERÊNCIA, APRESENTANDO SUAS CORRELAÇÕES. ILUSTRE.

- INDEPENDÊNCIA FUNCIONAL: a independência funcional é conseguida desenvolvendo se módulos com função “de um só propósito" e " a interações excessivas com outros módulos. Um software com módulos independentes é mais fácil de ser desenvolvido e mais fácil de ser mantido;
- COESÃO: medida da unidade funcional relativa de um módulo;