

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA DE CIÊNCIAS EXATAS E DA COMPUTAÇÃO.
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS.



Questionário

BRUNO CAMARGO MANSO

GOIÂNIA, GO
2021

BRUNO CAMARGO MANSO

Questionário

Matéria: Desenho de Aplicativos para Dispositivos Móveis

Orientador: Fabrício Schlag

GOIÂNIA, GO

2021

1. Conceitue:

a. Projeto Arquitetural

Segundo a ISO/IEEE 1471-2000, arquitetura é a organização fundamental de um sistema incorporada em seus componentes, seus relacionamentos com o ambiente, e os princípios que conduzem seu design e evolução. É uma representação da informação (ou parte dela) contida na arquitetura de forma que se adéque às necessidades de um ou mais interessados. Ela facilita o entendimento por parte do interessado, uma vez que vai filtrar e formatar a informação. O arquiteto pode usar as diferentes visões para lidar com complexidade. O termo arquitetura de software é usado para designar processo e produto

b. Arquitetura de Sistema

A Arquitetura de Sistemas de Informação tem o objetivo de analisar necessidades dos usuários em um sistema a ser desenvolvido. Não procurando se aprofundar em detalhes tecnológicos, mas sim se concentrar na necessidade do cliente, considerando as características do negócio do cliente.

c. Arquitetura de Software

Consiste na definição dos componentes de software, propriedades externas e relacionamentos com outros softwares. É a documentação da arquitetura de software do sistema que facilita a comunicação entre os stakeholders, registra as decisões iniciais de um projeto de alto-nível, permitindo o reuso do projeto de seus componentes e padrões.

d. Acoplamento

É o quanto uma classe depende da outra para funcionar. Quanto maior for esta dependência, maior o acoplamento. Um acoplamento forte pode elevar o custo de manutenção e o seu gerenciamento, uma mudança pode afetar toda uma cadeia de classes.

e. Coesão

A coesão prega que uma classe deve ter apenas uma única responsabilidade e realizá-la de maneira satisfatória. Uma classe não deve assumir responsabilidades além do estipulado. Classes de um projeto de boa coesão divide as responsabilidades entre as classes de forma que sua manutenção seja feita de maneira mais prática e que tenha sentido.

f. Interface

É a forma de comunicação entre duas entidades. De forma abstrata também pode-se definir como a comunicação entre uma entidade e o mundo exterior. Permite que uma entidade seja modificada sem afetar as entidades externas que interagem com ela. É importante no estudo da interação homem-máquina, em projetos de hardware e softwares. Interfaces do usuário se comunicam então com pessoas enquanto interfaces físicas se comunicam com componentes de hardware.

g. Componentes

Segundo a OMG, é “uma parte modular, possível de ser implantada e substituível de um sistema que encapsula implementação e expõe um conjunto de interfaces.” Em P.O.O um componente é a classe que implementa uma interface, autônoma em relação a outros componentes do sistema.

h. Conectores

Descrevem interações, estabelecendo restrições sobre a fluidez das mensagens trocadas pelos componentes. Suas informações são concentradas em uma interconexão tornando-a visível e reutilizável.

2. Explique o que são estilos arquiteturais?

É o mesmo que o termo “modelo arquitetural”. Um estilo define os tipos de elementos podendo aparecer em uma arquitetura bem como suas regras, reagindo às interconexões. Nesses estilos, estão contidos: Cliente-Servidor; “Em camadas”; Filtros e dutos (pipes and filters); Baseado em repositório; Orientado a eventos (publisher/subscriber); Transferência Representacional de Estado (REST); Objetos distribuídos; C2. Um estilo arquitetural representa um conjunto de escolhas de projeto. Estilos podem ser usados para descrever uma determinada arquitetura.

3. Como a UML se encaixa na criação de uma arquitetura de software?

É uma linguagem de modelagem, cujo papel é auxiliar a equipe de desenvolvimento a visualizar os diversos aspectos da aplicação, facilitando a compreensão do seu funcionamento e da sua arquitetura do software.

4. Qual o propósito da fase de Projeto dentro do processo de desenvolvimento de software? Diferencie-a da fase de Análise de Requisitos.

Um projeto se divide em 5 fases básicas: Iniciação, Planejamento, Execução, Monitoramento e Controle e Encerramento. Cada etapa é composta por ações específicas e cada fase do projeto complementa a anterior, sendo a base da posterior. Já o levantamento e análise de requisitos é um processo iterativo, com uma contínua validação de uma atividade para outra. É o processo de interagir com os stakeholders do sistema para descobrir seus requisitos. É nesta fase que se ganha compreensão e domínio sobre o projeto. Ou seja, é uma fase diferente das fases de projeto acima expostas.

5. Durante a realização de alterações em um produto de software, um sério problema que as organizações enfrentam é a propagação de efeitos colaterais de uma alteração para outras partes do sistema que, em princípio, não deveriam ser afetadas. Que táticas de projeto podem ser usadas para evitar esse problema?

As táticas utilizadas são o melhoramento contínuo e controle de versão.

6. Explique o que faz o arquiteto de software e qual é a expectativa em relação ao seu papel.

O Arquiteto de Software é responsável por garantir o seguimento das diretrizes de um projeto de desenvolvimento de software que normalmente são: qualidade dos sistemas, contexto organizacional, funcionalidade, usabilidade, performance, desempenho e baixo custo de investimento. Está envolvido mais com questões estratégicas do que técnicas de uma operação. Busca melhorar a performance pelo desenvolvimento de sistemas e aplicações com serviços. Habilidades interpessoais como liderança e gestão de pessoas são condições sine qua non de um bom arquiteto de software .

7. Podemos dizer que um projeto de arquitetura é um processo criativo em que se tenta estabelecer uma organização de sistema que satisfaça os requisitos funcionais e não funcionais do sistema? Justifique sua resposta.

Sim, é um processo criativo à medida que o Arquiteto de Software deve desenvolver soluções e entregá-las ao seu cliente respeitando os quesitos estabelecidos por ele e assim, transformando requisitos em códigos coesos e funcionais.

8. Para cada um dos estilos de arquitetura apresente a definição, componentes, e conectores:

a. Programa Principal e Subsistema

É hierarquizada, sendo que o programa principal utiliza subprograms que podem utilizar outros subprogramas

Componentes: são encapsulados em objetos. A comunicação e coordenação entre os componentes é realizado através da troca de mensagens (métodos dos objetos)

Conectores: Broadcast de eventos

b. Chamada remota de procedimento

É uma tecnologia de comunicação entre processos que permite a um programa de computador chamar um procedimento em outro espaço de endereçamento. Permite a divisão de programas em vários pedaços.

Conectores: TCP/IP

Componentes: Stub do cliente, Sistema operacional do cliente, Stub do servidor, Sistema operacional do servidor.

c. Cliente/servidor

Permite que dois componentes interajam através de solicitações Request/Reply

Componente: O componente de servidor fornece uma função ou serviço a um ou mais clientes, que iniciam os pedidos de serviço.

Conector: Request/Reply, usado para invocar serviços e enviar respostas.

d. Camadas

Organização hierárquica, cada camada oferece serviço a camada acima dela também serve como cliente da camada inferior.

Componentes: Relativo a cada camada.

Conectores: Definidos pelos protocolos que definem como uma camada se comunicará com a outra, mas limitando interações com camadas adjacentes.

e. Blackboard

É um tipo de repositório compartilhado que utiliza as informações dos seus clientes para gerar novo conhecimento o qual pode ser acessado de novo por seus clientes para refinar suas operações. Divide tarefas complexas em tarefas determinísticas e gera uma solução para ser compartilhada com os clientes.

Componente: De controle que monitora o blackboard e coordena os clientes de acordo ao estado do blackboard

Conectores: de escrita e leitura.

f. Repositório

Dados compartilhados são mantidos em um banco de dados central, que pode ser acessado por todos os subsistemas. Cada subsistema mantém seu próprio banco de dados e são passados explicitamente para outros subsistemas.

Componentes: interagem através de um repositório de dados compartilhados, porém não interagem diretamente entre eles, somente através do repositório.

Conectores: escrevem e lêem os dados.