

Grupo Handbook

Handbook de Questões de TI

comentadas para CONCURSOS

Além do gabarito

Volume 9

*Desenvolvimento de Software
Parte 2*

ESAF - Escola de Administração Fazendária

Prefácio

Este é o volume 9 da *Handbook de Questões de TI Comentadas para Concursos – Além do Gabarito*, que traz para você 50 questões elaboradas pela ESAF (Escola de Administração Fazendária).

A realização de concursos públicos no âmbito do Ministério da Fazenda e no de instituições conveniadas é uma das principais atividades da ESAF. Essa banca vem sendo a responsável pela organização de vários concursos de grande relevância na área de TI, como os da Receita Federal, Secretaria do Tesouro Nacional, Controladoria Geral da União, entre outros.

Portanto, para ajudá-lo ainda mais em sua preparação para concursos organizados pela ESAF, o Grupo Handbook de TI preparou para você o volume 9, que traz para você questões com grande ênfase na área de desenvolvimento de sistemas.

Nele você vai encontrar questões de Engenharia de Software, CMMI, MPS.BR, Orientação a Objetos, Governança de TI, Java, Segurança da Informação e muitos outros assuntos que sempre figuram nas provas para a área de desenvolvimento de sistemas.

Bons estudos,

Grupo Handbook de TI

Direitos Autorais

Este material é registrado no Escritório de Direitos Autorais (EDA) da Fundação Biblioteca Nacional. Todos os direitos autorais referentes a esta obra são reservados exclusivamente aos seus autores.

Os autores deste material não proíbem seu compartilhamento entre amigos e colegas próximos de estudo. Contudo, a reprodução, parcial ou integral, e a disseminação deste material de forma indiscriminada através de qualquer meio, inclusive na Internet, extrapolam os limites da colaboração. Essa prática desincentiva o lançamento de novos produtos e enfraquece a comunidade concurseira Handbook de TI.

A série *Handbook de Questões de TI Comentadas para Concursos – Além do Gabarito* é uma produção independente e contamos com você para mantê-la sempre viva.

Grupo Handbook de TI

Canais de Comunicação

O Grupo Handbook de TI disponibiliza diversos canais de comunicação para os concurreiros de TI.

Loja Handbook de TI

Acesse a nossa loja virtual em <http://www.handbookdeti.com.br>

Serviço de Atendimento

Comunique-se diretamente conosco através do e-mail faleconosco@handbookdeti.com.br

Twitter do Handbook de TI

Acompanhe de perto promoções e lançamentos de produtos pelo nosso Twitter <http://twitter.com/handbookdeti>

1. **Assuntos relacionados:** *Governança de TI, Organizational Project Management Maturity Model (OPM3), CMMI, Gerenciamento Eletrônico de Documentos (GED), COBIT, Portais Corporativos,*

Banca: ESAF

Instituição: *Agência Nacional de Águas (ANA)*

Cargo: *Analista Administrativo - Tecnologia da Informação e Comunicação / Desenvolvimento de Sistemas e Administração de Banco de Dados*

Ano: 2009

Questão: 1

O _____ é empregado na governança de recursos de Tecnologia da Informação (TI).

Assinale a opção que completa corretamente a frase acima.

- (a). OPM3
- (b). CMMI
- (c). GED
- (d). COBIT
- (e). Portal corporativo

Solução:

Entende-se por Governança de TI como um conjunto de estruturas e processos que visam garantir que a TI suporte e maximize os objetivos e as estratégias do negócio. Assim, a governança e o X (descobriremos quem é o X quando analisarmos as alternativas a seguir) devem suportar os interesses dos stakeholders internos e externos de acordo com as necessidades específicas de cada um: os de dentro da empresa que têm interesse em gerar valor a partir de investimentos de TI; os que prestam serviços de TI sejam eles internos (Departamento de TI) ou externos (fornecedores); e os que controlam riscos e responsabilidades.

Bom, agora nós analisaremos cada alternativa.

(A) ERRADA

OPM3 (Organizational Project Management Maturity Model, em português Modelo de Maturidade em Gerenciamento de Projetos Organizacionais) fornece um mecanismo que permite monitorar o progresso dos interesses estratégicos da organização através da execução precisa e bem sucedida dos projetos. Em outras palavras, ele possibilita que a empresa utilize a gestão de projetos para atingir suas metas nos prazos e dentro do orçamento.

Ao estudar o OPM3, você perceberá que ele é baseado no guia PMBOK (padrão global) e pode ser aplicado em organizações de diversos segmentos, tamanhos ou localizações geográficas. Notará também que o OPM3 desenvolve um ciclo continuado de aperfeiçoamento do conhecimento, da avaliação e da melhoria de forma que, conforme você se move pelo processo de melhoria, passa por uma mudança de pensamento que posicionará sua organização de modo a permitir que ela obtenha tanto ganhos imediatos como sucesso em longo prazo.

Você perceberá também que o OPM3 permite que as organizações avaliem seu nível de maturidade. No caso, existem 4 níveis (Padronizar, Medir, Controlar e Melhorar continuamente) para cada domínio de análise: Projetos que é o domínio, digamos, mais “baixo”;

Programas, domínio “intermediário”; e Portfólios, representado o domínio mais “alto”.

Candidato, lembre-se que maturidade é o desenvolvimento de sistemas e processos que são repetitivos por natureza e garantem uma alta probabilidade de que cada um deles seja um sucesso. Observe que, em nenhum momento fizemos menção à Governança de TI, o que nos dá condição para descartar esta alternativa.

(B) ERRADA

O CMMI (Capability Maturity Model Integration) é uma metodologia criada pela SEI (Software Engineering Institute) com o objetivo de substituir as diversas CMMs (Capability Maturity Model) desenvolvidas especificamente para cada área de atuação, tais como: desenvolvimento de sistemas, engenharia de software, aquisição de software e desenvolvimento de processos. Apesar das CMMs serem úteis, elas são excessivamente diversificadas e, por conta disso, foram consideradas problemáticas (redundância de conceitos, terminologia incomum, inconsistências, etc.)

De forma geral, o objetivo maior do CMMI é prover um conjunto de boas práticas para melhoria dos processos das organizações e a habilidade de gerenciar, desenvolver e manter os seus produtos (softwares). Essas práticas, já consideradas efetivas, são organizadas em uma estrutura que visa auxiliar a organização, em particular os gerentes de projetos, a estabelecer suas prioridades para a conquista de melhorias. Um guia para a implementação dessas melhorias também é fornecido pelo CMMI.

Candidato, você precisa saber as diferenças entre as formas de representação que o CMMI possui, uma contínua e outra por estágios (a existência de duas representações visa a oferecer flexibilidade para as organizações poderem utilizar diferentes meios para obterem melhorias de acordo com as suas realidades).

Enquanto que a representação contínua dá liberdade para as organizações utilizarem a melhor ordem entre as “melhorias” para que os objetivos de negócio sejam alcançados, a representação por estágios disponibiliza uma sequência pré-determinada para que as “melhorias” sejam obtidas. Essa sequência não pode ser desconsiderada pelas organizações que pretendem utilizar essa representação, pois cada estágio serve de base para o próximo.

Novamente, em nenhum momento fizemos menção à Governança de TI, o que nos permite eliminar esta alternativa.

(C) ERRADA

GED (Gerenciamento Eletrônico de Documentos) é uma tecnologia que provê um meio de facilmente gerar, controlar, armazenar, compartilhar e recuperar informações existentes em documentos. Os sistemas GED permitem que os usuários acessem os documentos de forma ágil e segura.

Como bem sabemos, os conhecimentos de uma empresa residem na massa de documentos que a mesma possui. É aqui que entra o GED, permitindo a preservação e a organização deste patrimônio, assegurando a informação necessária, na hora exata, para a pessoa certa.

Não é difícil intuir que, devido a sua capacidade de gerenciar documentos, a GED é uma

ferramenta indispensável para a Gestão do Conhecimento. Não é difícil concluir também que a alternativa está ERRADA.

(D) CORRETA

O Control Objective for Information and related Technology (COBIT) é um framework que fornece as melhores práticas para gerenciamento de processos de TI, atendendo as necessidades de gestão de uma organização, tratando os riscos do negócio, as necessidades de controle e as métricas de desempenho. Esse framework serve de suporte para implementar a Governança de TI numa organização garantindo que: a TI esteja alinhada com o negócio; a TI permita o negócio e maximize os benefícios; os recursos de TI sejam usados com responsabilidade; e que os riscos de TI sejam gerenciados apropriadamente.

Vale reforçar que o COBIT não é um padrão definitivo. Ele serve como apoio para a implementação de controles na Governança de TI, ditando o “o que fazer” e não se preocupando “como fazer”.

O COBIT foca em duas áreas chaves: fornece informações necessárias para suportar os objetivos e os requisitos do negócio; e trata as informações como sendo o resultado combinado de aplicações e recursos de TI que precisam ser gerenciados por processos de TI. Para isso, o COBIT agrupa os processos de TI em quatro áreas ou domínios:

1. Planejamento e Organização;
2. Aquisição e Implementação;
3. Entrega e Suporte;
4. Monitoração e Avaliação.

Candidato, lembre-se disso: o COBIT deve ser utilizado como um amplo guia pela direção e pelos responsáveis pelos processos de negócio da empresa, não restringindo apenas ao departamento de TI!

Portanto, a alternativa está CORRETA.

(E) ERRADA

Os portais corporativos são, comumente, apresentados como uma evolução das intranets. No entanto, os portais também podem ser vistos como uma aplicação web que agrega conteúdo, colaboração, conhecimento e aplicativos transacionais, todos em uma interface única. Exemplos de funcionalidades que os portais normalmente possuem são:

- busca e indexação;
- categorização de conteúdo;
- colaboração;
- personalização;
- comunidades;
- integração de sistemas;
- segurança.

A seguir, apresentamos alguns exemplos de portais corporativos (Candidato, decore o significado destas siglas!):

- **B2B (Business to Business)**: transações de comércio entre empresas. Por exemplo, um banco transferindo recursos para;
- **B2C (Business to Consumer)**: transações entre uma empresa e um consumidor. Por exemplo, portais da Submarino e da Lojas;
- **C2C (Consumer to Consumer)**: transações entre consumidores, intermediadas normalmente por uma empresa. Por exemplo, o sites de leilão Ebay e Mercado Livre;
- **B2G (Business to Government)**: transações entre empresa e governo, como, por exemplo, licitações e compras de fornecedores;
- **B2E (Business-to-Employee)**: portais que atendem aos empregados. Oferecem aos empregados uma interface única para que eles possam acompanhar as notícias da empresa, gerenciar seus dados pessoais, solicitar serviços, etc.

Portanto, nada relacionado à Governança de TI.

2. **Assuntos relacionados:** *Engenharia de Software, Modelo de Processo de Software, Desenvolvimento Evolucionário, Transformação Formal, Desenvolvimento em Cascata,*

Banca: ESAF

Instituição: Agência Nacional de Águas (ANA)

Cargo: Analista Administrativo - Tecnologia da Informação e Comunicação / Desenvolvimento de Sistemas e Administração de Banco de Dados

Ano: 2009

Questão: 8

O modelo de processo de software caracterizado por intercalar as atividades de especificação, desenvolvimento e validação, denomina-se

- (a). modelo de workflow.
- (b). modelo de fluxo de dados.
- (c). desenvolvimento evolucionário.
- (d). transformação formal.
- (e). modelo em cascata.

Solução:

(A) ERRADA

Um modelo de workflow é responsável por analisar e modelar todo o tramite de informações e atividades referentes a um processo de negócio de uma empresa para que possa ser definida uma forma organizada de execução das tarefas que compõem o processo. A automação do processo de negócio de uma empresa é o resultado final da modelagem de workflow, consistindo na interligação de tarefas executadas por participantes do processo ou até mesmo por sistemas computacionais definindo a ordem de execução e suas prioridades. Durante todo o desenvolvimento de um workflow várias etapas são executadas em uma determinada ordem. O ciclo de vida de um workflow (WFLC - Workflow Life Cycle) define a maneira como estas etapas são desenvolvidas em determinado tempo sobre todo o desenvolvimento do workflow. Para tanto, são definidos três atividades principais:

- **Construção e Estabilização:** este processo representa a fase de análise e modelagem do processo de negócio que a partir de uma linguagem gráfica é definido o Workflow, utilizando conceitos como processo, atividade, tarefa, ator, etc. Nesta fase é utilizado o processo de prototipagem para a detecção de erros de modelagem, por esse processo ser iterativo pode detectar e corrigir os erros de definição do modelo;
- **Detecção de Inconsistência:** nesta fase é gerada uma versão operacional referente à modelagem construída na fase anterior contendo todas as características e eventos do modelo. A versão é executada para que possa verificar o fluxo de informação que o modelo representa analisando informações de entrada do processo, recursos utilizados, participantes e resultado de saída do processo a fim de detectar inconsistência da análise da fase de construção do modelo. Ao encontrar anomalias no modelo o mesmo é submetido à estabilização novamente para melhorar a qualidade do Workflow;
- **Reengenharia de processo de Negócio:** os dados obtidos na fase anterior ao executar a versão operacional são usados a fim de verificar a adequação do modelo de processo de negócio para os objetivos da organização. Se este não for o caso, uma revisão do modelo do processo de negócio deve ser feita. O modelo atualizado é o ponto de partida de uma nova iteração dos processos de estabilização e depuração.

(B) ERRADA

O modelo de fluxo de dados é uma forma de representar os processos de um sistema, também, o fluxo das informações que entram nos processos e o resultado (saída) do processamento. Um fluxo de dados de um sistema pode ser visto como uma rede que interliga os processos mostrando a movimentação dos dados (entrada/saída). Este tipo de modelagem é importante para que o analista entenda todo o tramite das informações no sistema. A modelagem de fluxo de dados utiliza componentes para sua representação que são:

- **Processo:** é parte do sistema representa uma seqüência de operações (procedimento) que pode receber uma ou várias entradas, processar e retorna uma ou várias saídas;
- **Fluxo:** representa a movimentação dos dados, indicando onde as informações trafegam como entrada e onde são saídas. O Fluxo é representado por uma seta que indica a direção da movimentação dos dados;
- **Depósito:** Enquanto o fluxo representa as informações em movimento o depósito indica onde os dados estão armazenados (sem movimentação);
- **Entidade Externa:** É uma entidade (sistema externo) que se comunica com o sistema, a sistema externo pode ser uma pessoa, empresa, organização.

O modelo de fluxo de dado inicia em um nível de abstração alto e sofre processos de refinamento até chegar a um nível de detalhamento correspondente a sua necessidade. Para cada iteração de refinamento é criado um novo modelo atribuindo ao nome do documento um número correspondente a quantidade de iterações.

(C) CORRETA

O modelo evolucionário desenvolve todo o projeto de um sistema intercalando as suas atividades de especificação, desenvolvimento e validação. Este modelo trabalha com muitas versões dos sistemas que são apresentadas para o cliente e então refinadas, ou seja, as versões iniciais são construídas a partir de informações mais gerais e/ou abstratas e ao possa que são finalizadas e expostas para o cliente sofrem avaliações e melhoramentos. As atividades de desenvolvimento não são executadas separadamente para a construção do sistema, mas sim, concorrentemente para que ter uma resposta do cliente e um melhoramento/agregação das funcionalidades até chegar à versão final.

Há dois tipos de desenvolvimento evolucionário:

1. **Desenvolvimento exploratório:** tem como objetivo definir com o cliente os requisitos iniciais melhor compreendidos. O sistema sobre alterações à medida que o cliente detecta a necessidade de novas funcionalidades;
2. **Fazer protótipos descartáveis:** tem como objetivo refinar os requisitos dos sistemas baseando em protótipos para melhorar os requisitos que estão mal compreendidos.

(D) ERRADA

A transformação formal é um modelo de processo de software que utiliza de métodos formais para o desenvolvimento do software. Isto é, o processo de transformação formal executa várias etapas (atividades) que sequencialmente transformam os requisitos de software em um modelo matemático e posteriormente em um software. Este modelo tem como característica uma modelagem limpa, livre de erro. Isso é obtido através de modelos matemáticos que permitem uma verificação ou validação do modelo antes que o sistema seja implementado.

A primeira etapa é a definição dos requisitos juntamente com o cliente. Depois esta descrição do cliente é então detalhada saindo de um nível de abstração de usuário para o nível de especificação. A especificação é então transformada em um modelo formal detalhado (modelo matemático). Por fim, a implementação do modelo e teste utilizando alguma linguagem de programação.

(E) ERRADA

O modelo em cascata, também denominado clássico ou sequência linear, sugere uma abordagem sequencial para o desenvolvimento de software, que inicia pela análise e modelagem dos requisitos, projeto, codificação, teste até manutenção do sistema. A vantagem dessa abordagem sequencial é visto no gerenciamento do projeto, uma vez que suas atividades possuem pontos de inicialização e finalização, assim, em princípio esta modelagem é confiável e abrangem projeto de tamanhos variados. Por outro lado, por ter essa rigidez nas atividades de processo às etapas iniciais como levantamento de requisito e análise devem ser cuidadosamente implementadas, caso contrário, todas as outras etapas serão comprometidas.

As principais atividades de desenvolvimento são:

- **Análise:** as funcionalidades e as restrições do sistema são obtidas através de entrevistas com usuários do sistema. Posteriormente os requisitos são definidos e então é gerado um documento com a especificação das funcionalidade de forma detalhada;
- **Projeto de software:** O projeto de software reuni os requisitos (funcionais e não funcionais) em sistemas de hardware ou de software, para então, construir a arquitetura do sistema em um âmbito geral. O projeto de software envolve a identificação de todos os componentes e seus relacionamentos necessários para o desenvolvimento do sistema;
- **Implementação:** O projeto de software é construído gerando um comum conjunto de módulos (programas) ou um único de programas;
- **Testes de sistemas:** O sistema desenvolvido é submetido a testes que representam os requisitos desejados pelos usuários do sistema a fim de verificar erros e inconformidades do sistema;
- **Manutenção:** Depois dos testes o sistema é instado e colocado em funcionamento. A manutenção envolve corrigir erros que não foram descobertos na fase de testes e posteriormente de acordo com as necessidades dos clientes sofrer adições ou melhoramento de funcionalidades.

Concluindo, o único processo de software que intercala as atividades de especificação, desenvolvimento e validação produzindo em cada iteração uma nova versão do software é o modelo evolucionário de software. Portanto, a resposta correta é alternativa C.

3. **Assuntos relacionados:** *Web Services, Simple Object Access Protocol (SOAP), Universal Description, Discovery and Integration (UDDI), Web Services Description Language (WSDL),*

Banca: *ESAF*

Instituição: *Agência Nacional de Águas (ANA)*

Cargo: *Analista Administrativo - Tecnologia da Informação e Comunicação / Desenvolvimento de Sistemas e Administração de Banco de Dados*

Ano: *2009*

Questão: *9*

Na arquitetura de Web Services, o componente que compreende um serviço de diretórios para armazenamento de descrições de serviços é

- (a). o Protocolo de Acesso a Objetos Simples (SOAP).
- (b). a Descrição, Descoberta e Integração Universal (UDDI).
- (c). a Linguagem de Definição de Web Services (WSDL).
- (d). a Linguagem de Marcação Hiper-Texto (HTML).
- (e). a Arquitetura TCP/IP.

Solução:

(A) INCORRETA

O SOAP (originalmente do Inglês Simple Object Access Protocol) é um protocolo para intercâmbio de informação estruturada na implementação de Web Services. Como formato de mensagens, ele depende de XML e depende também de protocolos da camada de aplicação, como HTTP e RPC. Por ser um protocolo para troca de informação, o SOAP pode prover a fundação da pilha de protocolos de Web Services, sobre o qual uma infraestrutura maior pode ser construída. Sendo um protocolo de intercâmbio de informação e não um serviço de diretórios, esta opção não responde corretamente a questão.

(B) CORRETA

O UDDI (do Inglês Universal Description, Discovery and Integration) é um registro baseado em XML e independente de plataforma como um “meta serviço” (um serviço que serve serviços) para permitir consultas robustas em metadados. Ele fornece à indústria uma especificação para construção de registros de Web Services interoperáveis. O UDDI é um padrão mantido pela OASIS (Organization for the Advancement of Structured Information Standards) e permite a publicação de listagens de serviços, a descoberta de serviços e que aplicações interajam através da Internet. De todas as alternativas listadas nesta questão, essa é a única relacionada à publicação e descoberta de serviços e, portanto, é a alternativa correta.

(C) INCORRETA

A WSDL (do Inglês Web Services Description Language) é, como o próprio nome diz, uma linguagem para descrição de Web Services. Ela é uma linguagem baseada em XML usada para descrever serviços de rede como um conjunto de pontos que operam em mensagens que contém tanto informação orientada a documentos quando informação baseada em procedimentos. Um documento WSDL define serviços como coleções de pontos finais. Em WSDL,

a definição abstrata de pontos finais e mensagens é separada de sua implementação, o que permite o reuso de definições abstratas: mensagens, que são descrições abstratas dos dados transmitidos, e tipos de porta, que são coleções abstratas de operações.

Sendo uma linguagem para descrição de Web Services sem a capacidade de armazenamento de descrições de serviços, essa alternativa está incorreta.

(D) INCORRETA

A linguagem de marcação de hipertexto (HTML, do Inglês HyperText Markup Language) é uma linguagem de marcação para páginas web. Ela provê meios para a criação de documentos estruturados através da definição de semântica estrutural para cabeçalhos, parágrafos, listas, links e assim por diante.

Sendo uma linguagem para marcação de texto e sem qualquer característica dinâmica nativa (até a versão 4), e não fazendo parte, diretamente, do conjunto de tecnologias de Web Services, mas como uma forma de apresentação de documentos, HTML não representa uma resposta correta para essa pergunta.

(E) INCORRETA

A pilha de protocolos TCP/IP é um conjunto de protocolos de comunicação usada na Internet e outras redes similares. Seu nome advém de seus protocolos mais importantes: o de controle de transmissão (TCP) e o da Internet (IP), os dois primeiros a serem definidos no padrão.

Sendo um conjunto de protocolos para comunicação e não fazendo parte, explicitamente, do conjunto de tecnologias de Web Services, a arquitetura TCP/IP não é a solução correta para a questão.

Questao	Resposta
1	D
2	C
3	B
4	B
5	E
6	C
7	B
8	B
9	A
10	A
11	B
12	D
13	C
14	A
15	E
16	C
17	B
18	D
19	D
20	C
21	B
22	C
23	E
24	B
25	D
26	E
27	E
28	A
29	D
30	D
31	A
32	E
33	E
34	B
35	B
36	E
37	B
38	A
39	E
40	E
41	C
42	B
43	A
44	C
45	E
46	E
47	A
48	E
49	B
50	C

Índice Remissivo

- Árvore, 111
- Abstração, 65
- ACID, 89, 103
- Agrupamento de Dados, 21
- Algoritmos, 122
- Algoritmos de Ordenação, 118
- Atividades-Chaves, 43

- Banco de Dados, 17, 19, 23, 26, 87, 89, 98, 103, 105, 108
- Bits de Paridade, 38
- Business Intelligence, 105

- Categorias de Linguagens de Banco de Dados, 98
- Central de Serviços, 61
- Chave Assimétrica, 17, 41
- Chave Secundária, 17
- Chave Simétrica, 17, 41
- Checksum, 38
- Classes e Superclasses, 32
- Classificação de Dados, 21
- CMM, 96
- CMMI, 4, 51, 53
- COBIT, 4, 43
- Common Object Request Broker Architecture (CORBA), 114
- Criptografia, 17, 41
- Criptografia Assimétrica, 41
- Criptografia Simétrica, 41
- Cyclic Redundancy Check (CRC), 38

- Data Control Language (DCL), 87, 98
- Data Definition Language (DDL), 87, 98, 101
- Data Manipulation Language (DML), 87, 98, 101
- Data Mining, 21
- Data Query Language (DQL), 98
- Data Warehouse, 105
- Desenvolvimento de Sistemas, 51
- Desenvolvimento em Cascata, 8
- Desenvolvimento Evolucionário, 8
- Desenvolvimento Formal, 81
- Diagrama de Atividades, 91
- Diagrama de Caso de Uso, 91
- Diagrama de Colaboração, 91
- Diagrama de Componentes, 91
- Diagrama de Objetos, 91

- Distributed Component Object Model (DCOM), 114
- Distributed Computing Environment (DCE), 114

- Encapsulamento, 65
- Engenharia de Software, 8, 13, 15, 46, 53, 75, 81, 83, 85, 94
- Estruturas de Dados, 111

- Fila, 111

- Garbage Collection, 113
- Generalização, 15
- Gerenciamento de Nível de Serviço, 57
- Gerenciamento Eletrônico de Documentos (GED), 4
- Governança de TI, 4, 43, 57

- Heapsort, 118
- Herança, 32, 65
- Hierarquia de Generalização, 94

- Indexação de Registros, 108
- Integridade de Dados, 38
- Interface com Usuário, 71, 78
- ISO/IEC 12207, 51
- ITIL, 57, 61

- Java, 32, 34, 36, 68, 71, 73, 113, 122
- Java Remote Method Invocation (Java RMI), 114
- JavaServer Pages (JSP), 36

- Linguagem de Modelagem, 77
- Linguagens de Programação, 32, 34, 36, 68, 71, 73, 122
- Lista Encadeada, 111

- MA-MPS, 46
- MN-MPS, 46
- Modelagem de Dados, 17
- Modelagem de Dependência entre Dados, 21
- Modelo de Maturidade, 51
- Modelo de Processo de Software, 8, 81, 83
- Modelo Entidade-Relacionamento, 23
- Modelo Espiral, 83
- Modelo Objeto-Relacionamento, 23
- Modelo Relacional, 23
- Modificadores de Acesso, 32
- MPS.BR, 46, 51
- MR-MPS, 46

Níveis de Maturidade, 96
Nível de Abstração, 19
Nível de Visão, 19
Nível Físico, 19
Nível Lógico, 19

On-line Analytical Processing, 105
On-Line Transaction Processing, 105
Organizational Project Management Maturity Model (OPM3), 4
Orientação a Objeto, 32, 63, 65
Otimizador de Consultas, 101

Pacotes Java, 71, 73
Padrão Facade, 75
Padrões de Projeto, 75
Pilha, 111
Polimorfismo, 65
Portais Corporativos, 4
Programação, 31, 63, 65, 113
Programação Estruturada, 31
Projeto de Serviço, 57

Qualidade de Software, 51, 53

Remote Procedure Call (RPC), 114
Requisito de Domínio, 13
Requisito de Software, 13
Requisito Funcional, 13
Requisito Não-Funcional, 13
Requisitos de Sistema, 85
Requisitos de Usuário, 85

Segurança da Informação, 17, 38
SGBD, 101
Simple Object Access Protocol (SOAP), 11
Sistemas de Informações Gerenciais, 124
Sistemas Distribuídos, 114
SQL, 26, 87, 98
Sumarização de Dados, 21
Superchave, 17

Tipos de Métodos, 63
Transaction Control Language (TCL), 98
Transformação Formal, 8
Tratamento de Exceção em Java, 34

UML, 15, 77, 91, 94
Universal Description, Discovery and Integration (UDDI), 11
Usabilidade, 78

Web Services, 11
Web Services Description Language (WSDL), 11
XML, 28