

MINISTÉRIO DAS COMUNICAÇÕES  
AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES  
(ANATEL)

CARGO 3: ANALISTA ADMINISTRATIVO – ESPECIALIDADE:  
DESENVOLVIMENTO DE SISTEMAS DE INFORMAÇÃO

PROVA DISCURSIVA  
DISSERTAÇÃO

APLICAÇÃO: 14/9/2014

**PADRÃO DE RESPOSTA**

Espera-se que o candidato redija texto dissertativo acerca da importância da segurança da informação nas organizações.

**Papel da informação nas empresas**

A informação é um ativo essencial aos negócios de uma organização, por isso deve ser adequadamente protegida. Como o ambiente dos negócios está cada vez mais interconectado, o resultado desse significativo aumento de interconectividade é a exposição da informação a uma grande variedade de ameaças e vulnerabilidades.

**Segurança da informação: definição e formas de implantação nas organizações**

A segurança da informação é um sistema de proteção da informação contra vários tipos de ameaças. Os mecanismos de segurança da informação visam garantir a continuidade do negócio, minimizar os riscos bem como maximizar o retorno sobre os investimentos e as oportunidades de negócio.

A segurança da informação é obtida por meio da implementação de um conjunto de controles adequados à proteção da informação, como políticas, processos, procedimentos, estruturas organizacionais e funções de *software* e *hardware*.

**Tipos de ameaças à segurança da informação**

As organizações, seus sistemas de informação e redes de computadores são expostos a diversos tipos de ameaças à segurança da informação, como, por exemplo, fraudes eletrônicas, espionagem, sabotagem, vandalismo, incêndio e inundação. Os danos provocados pela ação de códigos maliciosos, *hackers* e ataques de *denial of service* estão se tornando cada vez mais comuns.

MINISTÉRIO DAS COMUNICAÇÕES  
AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES  
(ANATEL)

CARGO 3: ANALISTA ADMINISTRATIVO – ESPECIALIDADE:  
DESENVOLVIMENTO DE SISTEMAS DE INFORMAÇÃO

PROVA DISCURSIVA  
QUESTÃO 1

APLICAÇÃO: 14/9/2014

**PADRÃO DE RESPOSTA**

Espera-se que o candidato redija texto dissertativo acerca do *test driven development* (TDD).

**Características básicas do TDD**

A característica básica do TDD (*test driven development*) é que, no desenvolvimento de um *software*, os testes são criados antes da escrita dos códigos que permitirão o uso desses testes. É uma abordagem evolutiva na qual o desenvolvedor escreve o teste antes de escrever o código funcional para satisfazer esse teste.

**Vantagens do emprego do TDD em relação a outras metodologias ágeis (são listadas abaixo somente as três principais)**

Entre as vantagens do emprego do TDD em relação a outras metodologias ágeis, incluem-se a oferta de programa de maior qualidade por estar direcionado às expectativas do cliente, já que o trabalho do desenvolvedor é feito a partir de considerações acerca de como as funcionalidades serão utilizadas por esse cliente; a possibilidade de se testar todo o código desenvolvido, o que dá maior confiabilidade ao sistema; e a geração, via de regra, de um código mais modularizado, flexível e extensível, pois a metodologia requer que os desenvolvedores pensem no *software* como pequenas unidades que podem ser reescritas, desenvolvidas e testadas de forma independente e integradas em momento posterior.

**Princípios do XP apoiados pelo TDD e associação entre os procedimentos**

Um princípio do XP (*extreme programming*) a que o TDD está associado é o do Teste Primeiro. O TDD pode apoiar esse princípio por fornecer detalhes para a realização dos testes de unidade e de funcionalidade, que são importantes e necessários. Além disso, o TDD é útil no que diz respeito ao princípio do XP da Refatoração, pois confere ao programador maior segurança para identificar e remover o código duplicado, e permite, assim, a melhoria contínua do programa.

Outros princípios do XP poderão ser considerados, embora não citados na literatura como os de maior destaque, desde que justificados a sua associação com o TDD de forma coerente e técnica. Exemplo: Integração Contínua, pois os testes escritos durante o TDD são utilizados para realização de testes de regressão à cada ciclo de integração do XP.

MINISTÉRIO DAS COMUNICAÇÕES  
AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES  
(ANATEL)

CARGO 3: ANALISTA ADMINISTRATIVO – ESPECIALIDADE:  
DESENVOLVIMENTO DE SISTEMAS DE INFORMAÇÃO

PROVA DISCURSIVA  
QUESTÃO 2

APLICAÇÃO: 14/9/2014

**PADRÃO DE RESPOSTA**

Espera-se que o candidato redija texto dissertativo acerca da notação de modelagem de processos BPMN (*business process modeling notation*).

**Vantagens da utilização da BPMN para a modelagem de processos**

A BPMN é uma notação mantida pela OMG (*Object Management Group*), mas de padrão aberto, o que permite que qualquer indivíduo encaminhe sugestões ao grupo mantenedor. Ainda pelo fato de se tratar de padrão aberto, qualquer desenvolvedor pode construir ferramentas sem a obrigação de pagamento de taxas, o que resulta em vasta gama de ferramentas disponíveis — algumas de forma gratuita. O padrão aberto também permite a comunicação com outras linguagens, como, por exemplo, a XPDL, que é uma linguagem de descrição de fluxos de trabalho (*workflow*) e a WSBPEL, linguagem de execução de processos BPM.

Outra vantagem da BPMN é sua flexibilidade, que permite que um mesmo cenário (processo) possa ser representado de várias formas e implementado da maneira mais conveniente para a organização.

**Principais elementos básicos de modelagem BPMN**

Os elementos básicos da BPMN são os que permitem uma modelagem simplificada em BPMN. O principal componente de um processo é o trabalho realizado. **Atividade** (*activity*) é o elemento que representa o trabalho, e é simbolizado por um retângulo. Para representação de tomadas de decisão ou alterações no fluxo natural, o uso de **gateways** pode ser representado por losangos. Ao longo do processo, podem ocorrer situações que precisam ser destacadas, como, por exemplo, o início e o fim do processo, e a entrega de um documento. Essas situações são denominadas **eventos**, representados por círculos (o início do processo é representado por círculo de borda fina, o fim do processo, por círculo de borda grossa, e os eventos intermediários, por círculo com borda dupla). Para determinar os atores do processo, são utilizadas **piscinas** (*pools*) e **raias** (*lanes*). A piscina representa o processo como um todo, e é simbolizada pelo retângulo externo do processo. As raias são divisões da piscina e representam cada ator do processo, sendo retratada pelos retângulos internos. Para conectar os elementos e indicar a sequência entre eles, são utilizados **fluxos de sequência**, representados por setas.

MINISTÉRIO DAS COMUNICAÇÕES  
AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES  
(ANATEL)

CARGO 3: ANALISTA ADMINISTRATIVO – ESPECIALIDADE:  
DESENVOLVIMENTO DE SISTEMAS DE INFORMAÇÃO

PROVA DISCURSIVA  
QUESTÃO 3

APLICAÇÃO: 14/9/2014

**PADRÃO DE RESPOSTA**

Espera-se que o candidato redija texto dissertativo acerca da versão atual do MVC (*model-view-controller*) e dos padrões de projeto que favorecem o desacoplamento e a escalabilidade do *software*.

**Partes constituintes do MVC**

O modelo arquitetural MVC é composto das seguintes partes:

*Model*: ou modelo é a camada que contém a lógica da aplicação, e é responsável pelas regras de negócio para sistemas persistentes.

*View*: ou visão é a camada de apresentação com usuário, é a interface que proporcionará a entrada de dados e a visualização de respostas geradas.

*Controller*: é o elemento que controla e coordena o envio de requisições feitas entre a visão e o modelo. O *controller* define o comportamento da aplicação.

**Padrão *observer***

A arquitetura MVC utiliza padrões de projetos em suas camadas. O *model* pode fazer o uso do padrão *observer*, que separa a visão do estado de um objeto do próprio objeto, permitindo que sejam fornecidas visões alternativas mantendo os objetos interessados constantemente informados sobre suas mudanças de estado. O uso do padrão *observer* mantém o modelo desacoplado totalmente independente das visualizações e dos controladores, o que permite a escalabilidade da solução, uma vez que múltiplas visualizações podem ser feitas ao mesmo tempo.

**Padrão *composite***

O padrão *composite* também contribui para a escalabilidade e desacoplamento do *software*. Nesse caso, o controlador determina a visualização percebida na camada de visão do usuário. As respostas e os dados gerados nessa camada de visão são enviados para o *controller* (ou controlador), que controla e coordena as ações a serem realizadas conforme o padrão de projeto *composite*.

Por exemplo, um painel de controle de botões pode ser implementado como uma visão complexa contendo visões encaixadas compostas de botões. A abordagem MVC suporta visões encaixadas com a classe *CompositeView*, uma subclasse de *View*. Os objetos de *CompositeView* funcionam exatamente como objetos de *View*; uma visão composta pode ser usada em qualquer lugar que uma visão possa ser usada, mas ela também contém e administra visões encaixadas.