



FUNDAÇÃO UNIVERSIDADE DE BRASÍLIA

CARGO 2: ANALISTA DE TECNOLOGIA DA INFORMAÇÃO

Prova Discursiva

Aplicação: 13/10/2013

PADRÃO DE RESPOSTA

Quesitos 1 e 2

O candidato deve listar e dissertar sobre as atividades comuns entre os processos de desenvolvimento de *software* existentes e, em seguida, descrever um padrão de desenvolvimento. Essas atividades e suas respectivas descrições são:

Análise econômica

Visa a estabelecer se o projeto de *software* gerará lucro e se a receita gerada será o suficiente para cobrir os custos.

Especificação – extração dos requisitos de um cliente

A especificação é a tarefa de descrever precisamente o *software* que será escrito, preferencialmente de uma forma matematicamente rigorosa. Na prática, somente especificações mais bem sucedidas foram escritas para aplicações bem compreendidas e afinadas que já estavam bem desenvolvidas, embora sistemas de *software* de missão crítica sejam frequentemente bem especificados antes do desenvolvimento da aplicação. Especificações são mais importantes para interfaces externas que devem permanecer estáveis.

Arquitetura de *software*

A arquitetura de um sistema de *software* remete a uma representação abstrata daquele sistema. Arquitetura é concernente à garantia de que o sistema de *software* irá ao encontro de requisitos do produto, como também assegurar que futuros requisitos possam ser atendidos. A etapa da arquitetura também direciona as interfaces entre os sistemas de *software* e outros produtos de *software*, como também com o *hardware* básico ou com o sistema operacional.

Implementação (ou codificação)

A transformação de um projeto para um código deve ser a parte mais evidente do trabalho da engenharia de *software*, mas não necessariamente a sua maior porção.

Teste

Teste de partes do *software*, especialmente onde tenha sido codificado por dois ou mais engenheiros trabalhando juntos, é um papel da engenharia de *software*. Diversas atividades de testes são executadas a fim de se validar o produto de *software*, testando cada funcionalidade de cada módulo, levando em consideração a especificação feita na fase de projeto, onde o principal resultado é o relatório de testes, que contém as informações relevantes sobre erros encontrados no sistema, e seu comportamento em vários aspectos.

Documentação

Uma importante tarefa é a documentação do projeto interno do *software* para propósitos de futuras manutenções e aprimoramentos. As documentações mais importantes são as das interfaces externas.

Suporte e treinamento de *software*

Uma grande porcentagem dos projetos de *software* falham pelo fato de o desenvolvedor não perceber que não importa quanto tempo a equipe de planejamento e desenvolvimento irá gastar na criação do *software* se ninguém da organização irá usá-lo. As pessoas ocasionalmente resistem à mudança e evitam aventurar-se em áreas pouco familiares. Então, como parte da fase de desenvolvimento, é muito importante o treinamento para os usuários de *software* mais entusiasmados, alternando o treinamento entre usuários neutros e usuários favoráveis ao *software*. Usuários terão muitas questões e problemas de *software*, os quais conduzirão para a próxima fase.

Manutenção

A manutenção e melhoria de *software* lidam com a descoberta de novos problemas e requisitos. Estas podem tomar mais tempo que o gasto no desenvolvimento inicial do mesmo. Não somente pode ser necessário adicionar códigos que combinem com o projeto original, mas determinar como o *software* trabalhará em algum ponto depois

da manutenção estar completa. Isto pode requerer um significativo esforço por parte de um engenheiro de *software*. Cerca de dois terços de todos os engenheiros de *software* trabalham com a manutenção, mas estas estatísticas podem estar incorretas. Uma pequena parte destes trabalha na correção de erros. A maioria das manutenções visa ampliar os sistemas para novas funcionalidades, as quais, de diversas formas, podem ser consideradas um novo trabalho. Analogamente, cerca de dois terços de todos os engenheiros civis, arquitetos e construtores trabalham com manutenção de uma forma similar.

Quesito 3 – Padrão de desenvolvimento de *software*

O processo de desenvolvimento de *software* tem sido objetivo de vários padrões, que visam a certificação de empresas como possuidoras de um processo de desenvolvimento, o que garantiria certo grau de confiança aos seus contratantes. São padrões existentes atualmente:

CMMI (anteriormente CMM)
ISO/IEC 15504 (anteriormente SPICE)
ISO/IEC 12207
[[MR-MPS]]