



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Engenharia de Software

AD1 2º semestre de 2019.

1. Explique a história do surgimento da Engenharia de Software. Use as seguintes perguntas como referência para seu texto: Quando a Engenharia de Software surgiu? Porque se percebeu, naquele momento, que a Engenharia de Software era necessária? Porque relacionar o desenvolvimento de software com as Engenharias? Como o processo geral das Engenharias – análise, síntese e correções – foi trazido para o contexto do desenvolvimento de software? Esperamos algo em torno de 20 linhas de resposta, com consulta aos livros do curso. (Valor 3,0 pontos).

A resposta a esta pergunta pode seguir diversos caminhos, mas esperamos que os alunos citem pelo menos as seguintes informações: (a) o termo Engenharia de Software foi cunhado no final da década de 1960, face às dificuldades de construção de software, que já eram conhecidas naquela época; (b) nesta época, os custos do hardware começaram a cair e o custo do trabalho de desenvolvimento de software passou a ser relevante e se destacar – daí foi necessário estudar formas de tornar este trabalho mais barato e previsível; (c) as Engenharias usam o conhecimento científico para tornar certos processos de construção economicamente viáveis – como o objetivo era justamente formar um corpo de conhecimento para viabilizar o desenvolvimento de software em larga escala e custo mais reduzido, a Engenharia de Software surgiu como um ramo de ciência que segue a filosofia geral das Engenharias; (d) os modelos de ciclo de vida de desenvolvimento de software seguem o processo geral das Engenharia, composto de análise (análise de viabilidade, levantamento de requisitos), síntese (projeto, codificação) e correções (testes e manutenção).

2. Com base no que você aprendeu sobre modelos de ciclo de vida, considere que você é o responsável por dois projetos, com equipes totalmente separadas. O primeiro projeto é um sistema de informação de média escala (imagine algo em torno de 50 mil linhas de código) com uma equipe de 3 pessoas e um domínio conhecido. O segundo é um sistema de engenharia, com cálculos e algoritmos complexos e grandes volumes de dados, cujo domínio é novo para a empresa e para a equipe. Que modelo de ciclo de vida serviria como base para cada projeto? Justifique sua resposta. Esperamos pelo menos 10 linhas de resposta, com consulta aos livros do curso. (Valor: 2,0 pontos)

Novamente, esta pergunta pode ser respondida de diversas formas. Esperamos que o aluno reconheça que o primeiro projeto é mais simples e, portanto, pode utilizar um ciclo de vida mais simples. Neste caso, o ciclo de vida em cascata parece uma boa opção. Por outro lado, o segundo projeto é mais complexo e exige um ciclo de vida que permita maior adaptação e uma adição incremental de requisitos e detalhes. Neste caso, um ciclo de vida incremental ou em espiral é mais adequado.

3. Para a construção do software, uma atividade importante é a elaboração de um documento descrevendo os requisitos do software. Desta forma, explique o que são requisitos, que tipos de requisito devem ser normalmente descritos para apoiar a construção do software e, considerando os diferentes ciclos de vida que você descreveu na questão 2, indique como as atividades relacionadas a requisitos podem ser realizadas neles. (Valor: 3,0 pontos)

Requisitos são declarações do que um sistema deve fazer, que dependem das tarefas cuja execução será suportada pelo sistema e das necessidades específicas da organização que o utilizará. São características ou funções que os sistemas devem possuir para que atendam aos objetivos desta organização e resolvam um conjunto de problemas. Os principais tipos de requisitos são funcionais (descrições das funções que o sistema deve prover para o usuário, ligados ao domínio do problema) e não-funcionais (restrições impostas sobre as funções oferecidas pelo sistema, ligadas ao domínio da solução). A principal atividade ligada aos requisitos é a elicitação (ou análise) de requisitos. Ela é a primeira atividade do ciclo de vida em cascata, precedendo o projeto (*design*) do software. Nos ciclos incrementais, ela também é a primeira atividade, mas é executada diversas vezes, seguindo os diversos incrementos do ciclo de vida. Outra atividade intimamente relacionada aos requisitos é a atividade de testes, que usa a descrição dos requisitos como referência para validar a implementação do software.

4. Explique o que é acoplamento na disciplina de projeto de software. Porque dizemos que um bom projeto visa reduzir o acoplamento entre os módulos de um programa? (Valor: 2,0 pontos)

O acoplamento é uma medida de interconexão entre os módulos de um sistema, que depende da complexidade da interface entre estes módulos. Bons projetos de software buscam reduzir o acoplamento entre os módulos que os compõem para evitar que erros gerados em um módulo se propaguem para os demais, para reduzir o conhecimento necessário a um desenvolvedor quando for aplicar uma alteração sobre as funcionalidades providas pelo sistema e para facilitar o reaproveitamento dos módulos em outros programas.