

Modelagem

A abstração do sistema de software através de modelos que o descrevem é um poderoso instrumento para o entendimento e comunicação do produto final que será desenvolvido.

A maior dificuldade nesta atividade está no equilíbrio (tradeoff) entre simplicidade (favorecendo a comunicação) e a complexidade (favorecendo a precisão) do modelo.

Para a modelagem podemos citar 3 métodos:

Análise estruturada, criada por Gane & Searson;

Análise Essencial, criada por Palmer & McMenamin e Ed. Yourdon;

UML, criada por Grady Booch, Ivar Jacobson & Jaimes Rumbaugh (veja exemplos). É, hoje, o método mais comum.

Ferramentas, tecnologias e práticas

A engenharia de software aborda uma série de práticas e tecnologias, principalmente estudadas pela ciência da computação, enfocando seu impacto na produtividade e qualidade de software.

Destacam-se o estudo de linguagem de programação, banco de dados e paradigmas de programação, como:

Programação estruturada

Programação funcional

Programação orientada a objetos

Componentes de Software

Programação orientada a aspecto



Ferramentas

Outro ponto importante é o uso de ferramentas CASE (do inglês Computer-Aided Software Engineering). Essa classificação abrange toda ferramenta baseada em computadores que auxiliam atividades de engenharia de software, desde a análise de requisitos e modelagem até programação e testes.

Os ambientes de desenvolvimento integrado (IDEs) têm maior destaque e suportam, entre outras coisas:

Editor

Compilador

Debug

Geração de código

Modelagem

Deploy

Testes não automatizados

Testes automatizados

Refatoração (Refatoring)

Gestão de Riscos nos projectos de Software

Uso da Prototipagem na Eng. de Requisitos