



UnB

Departamento de
Ciência da Computação

Projeto OO

Há duas maneiras de fazer projeto de um sistema de software.
Uma delas é fazê-lo tão simples que não há deficiências.
E a outra é fazê-lo tão complexo que não há deficiências óbvias.
O primeiro método é de longe o mais difícil.
C. A. Hoare

Edison Ishikawa, D. Sc.



Introdução

- Objetivo
 - Apresentar o conceito de Projeto OO

Sumário

- Introdução
- Desenvolvimento
- Referências



Introdução

- Não existe uma fronteira clara entre Análise e Projeto
- “A *análise vai se transformando em projeto* à medida que o desenvolvimento evolui”
- Mas são artefatos da análise
 - Modelo de Casos de Uso
 - Modelo de Classes de Análise
 - Preocupação maior é em identificar as classes
- Modelos feitos na análise apenas esclarecem o problema a ser resolvido, mas são insuficientes para se ter uma visão completa do sistema para a implementação

Introdução

- No projeto é que são definidas as soluções
- O objetivo é encontrar alternativas para que o sistema atenda aos requisitos funcionais, respeitando os requisitos não-funcionais

Projeto

- Principais atividades
 - Detalhes dos aspectos dinâmicos do sistema
 - Refinamentos dos aspectos estáticos e estruturais do sistema
 - Detalhamento da arquitetura do sistema
 - Definição das estratégias para armazenamento, gerenciamento e persistência dos dados manipulados pelo sistema
 - Realização do projeto da interface gráfica com o usuário
 - Definição dos algoritmos a serem utilizados na implementação

Detalhamento dos aspectos dinâmicos

- Modelo de interações
- Modelo de estados
- Modelo de atividades

Refinamento dos aspectos estáticos e estruturais

- Modelo de classes de projeto
 - O detalhamento dos aspectos dinâmicos do sistema gera material para refinar os aspectos estáticos e estrutural definidos no modelo de classes de análise
 - Detalhes têm o objetivo de obter um modelo suficientemente completo para que a implementação de classes possa ser feita a partir dele

Projeto de arquitetura

- Como o sistema pode ser decomposto em vários subsistemas.
 - Disposição das camadas de software que compõem o mesmo
- Aplicações distribuídas
 - Como o sistema deve ser fisicamente distribuído
 - Definição de meios pelo qual objetos em máquinas diferentes se comunicam
- Arquitetura – forma como um sistema está disposto e subdividido, física e logicamente.



Persistência de objetos

- Objetos transientes
 - Tempo de vida é o de uma sessão de uso do sistema
 - Desaparecem quando o processo que os criou termina
- Objetos persistentes
 - Devem existir por várias sessões do sistema
 - Como manter suas informações entre sessões de uso
 - Necessidade de controlar aspectos relativos a objetos persistentes
 - Como as transações são controladas
 - Quando e como devem ser enviados para mecanismos de armazenamento persistentes
 - Quando e como devem ser lidos de mecanismos de armazenamento persistentes
 - Quando e como os objetos persistentes são removidos



Projeto de interface gráfica com o usuário

- Associação entre casos de uso e atores implicam em necessidades de interface
- Ator humano
 - Telas (formulários), relatórios, etc...
 - Alta usabilidade e facilidade de operação
 - Padronização de cores, de mensagens de erro, das dimensões dos controles gráficos, formatação para entrada de dados, etc...
 - Interface Homem-Máquina
- Ponto crítico para o projeto, determinand o seu sucesso ou fracasso

Projeto de Algoritmos

- Aspectos que influenciam na escolha dos algoritmos
 - Complexidade computacional
 - Facilidade de entendimento
 - Flexibilidade

Referências

- Ferramentas de modelagem visual
 - Rational Rose (www.rational.com)
 - ASTAH Community (astah.net/editions/community)
- Livros
 - The Unified Modeling Language User Guide, Grady Booch et al
 - Engenharia de software – uma abordagem profissional, Roger S. Pressman
 - Princípios de análise e projetos de sistemas com UML, Eduardo Bezerra
- Especificações
 - www.omg.org



Dúvidas

