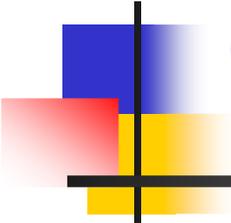


FERRAMENTA PARA CÁLCULO DE MÉTRICAS PARA SOFTWARES ORIENTADOS A OBJETOS



Acadêmica: Patrícia R.R. da Silva

Orientador: Marcel Hugo



ROTEIRO

- Introdução
- Métricas
- Métricas para Orientação a objetos
- Desenvolvimento do trabalho
- Conclusões



INTRODUÇÃO

- Justificativas para utilização de métricas
 - Gerenciar produtividade
 - Indicar a qualidade do produto
 - Formar uma linha básica de estimativas
 - Melhorar os esforços de desenvolvimento
- Utilização de técnicas OO para melhorar a qualidade do produto
- As métricas tradicionais não se aplicam adequadamente aos softwares OO



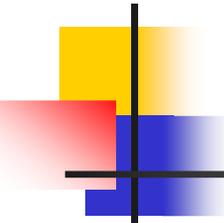
Objetivo

- Protótipo de ferramenta que calcule métricas para softwares Orientados a Objetos codificados em Delphi.



Métricas

- O que são métricas?
- As métricas podem ser divididas, sob o ponto de vista de medição em:
 - Diretas
 - Indiretas
- Ou ainda, sob o ponto de vista da aplicação:
 - Métricas de produtividade
 - Métricas de qualidade
 - Métricas técnicas



Pré-requisitos para utilização de Métricas

- Objetivos definidos
- Métricas devem ser simples
- Métricas devem ser objetivas
- Métricas devem ser efetivas no custo



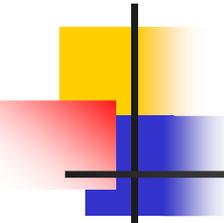
Métricas – Observações

- Uma única métrica normalmente não é útil
- As métricas são quase sempre interrelacionadas
- Métricas para serem úteis devem ser coletadas regularmente e de maneira automática
- As métricas podem ser correlacionadas com a realidade, ajudando a tomar decisões coerentes
- Para comparar as métricas deve-se conhecer bem as diferenças e similaridades entre as pessoas, processos e produtos



Métricas para orientação a objetos

- O desenvolvimento OO ainda não dispõe de métricas precisas e bem entendidas.
- A Orientação a Objetos surge como uma possibilidade para a melhoria da qualidade e produtividade
- A orientação a objetos requer uma abordagem diferente



Diferenças das métricas OO

- Métricas em O.O. são diferentes por causa da:
 - localização
 - encapsulamento
 - ocultamento das informações
 - herança
 - técnicas de abstração de objetos



Métricas OO devem avaliar

- Eficiência
- Complexidade
- Entendimento
- Reusabilidade
- Testabilidade/Manutenção



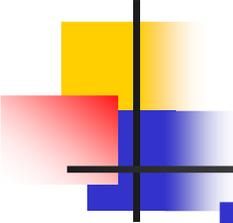
Categorias Métricas OO

- Métricas de Análise
 - Porcentagem de classes-chave
 - Número de cenários de utilização
- Métricas de projeto
- Métricas de construção



Métricas de Projeto

- Contagem de métodos
- Métodos ponderados por classe (WMC)
- Resposta de uma classe (RFC)
- Profundidade da árvore de herança (DTI)
- Número de filhos (NOC)
- Falta de coesão (LCOM)
- Acoplamento entre objetos (CBO)



Métricas de construção

- Tamanho do método
 - Linhas de código (LOC)
 - Média do tamanho dos métodos
- Percentual comentado
- Tamanho da classe
 - Quantidade de métodos de instância públicos
 - Quantidade de métodos de instância
 - Quantidade de atributos por classe
 - Média de atributos por classe
 - Quantidade de métodos de classe



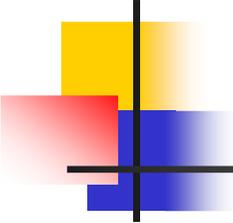
Métricas de construção - cont

- Quantidade de métodos sobrescritos por uma subclasse
- Quantidade de métodos herdados por uma subclasse
- Quantidade de métodos adicionados por uma subclasse
- Índice de especialização



DESENVOLVIMENTO DO TRABALHO

- Principais Requisitos
 - Extrair informações de classes, métodos e atributos do código fonte
 - Calcular as métricas selecionadas



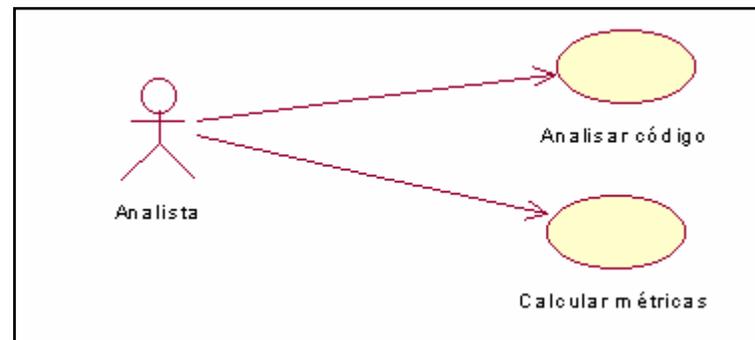
Métricas implementadas

- Métricas de projeto e construção explicadas anteriormente

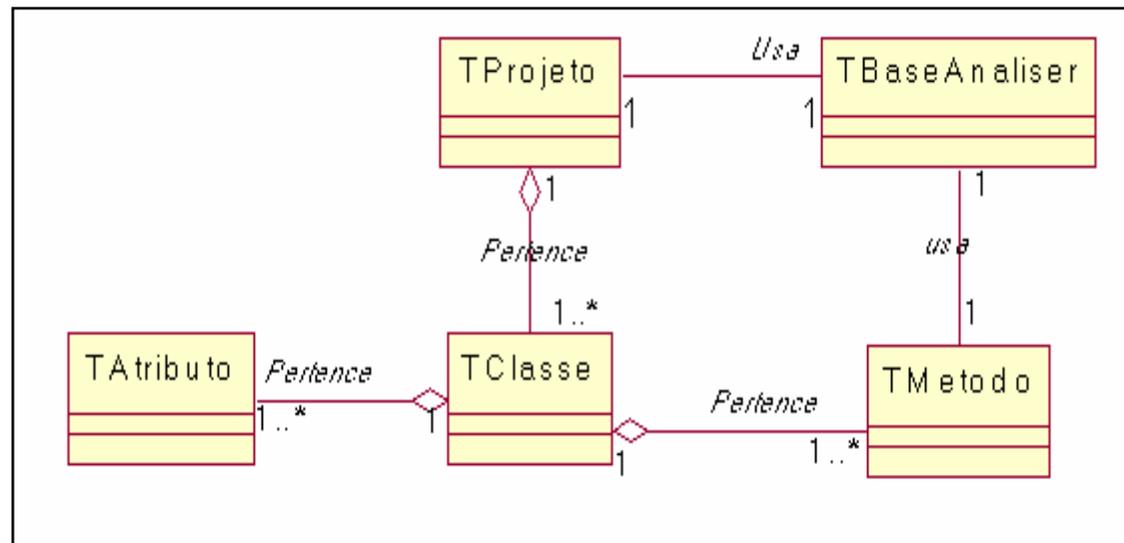
Especificação

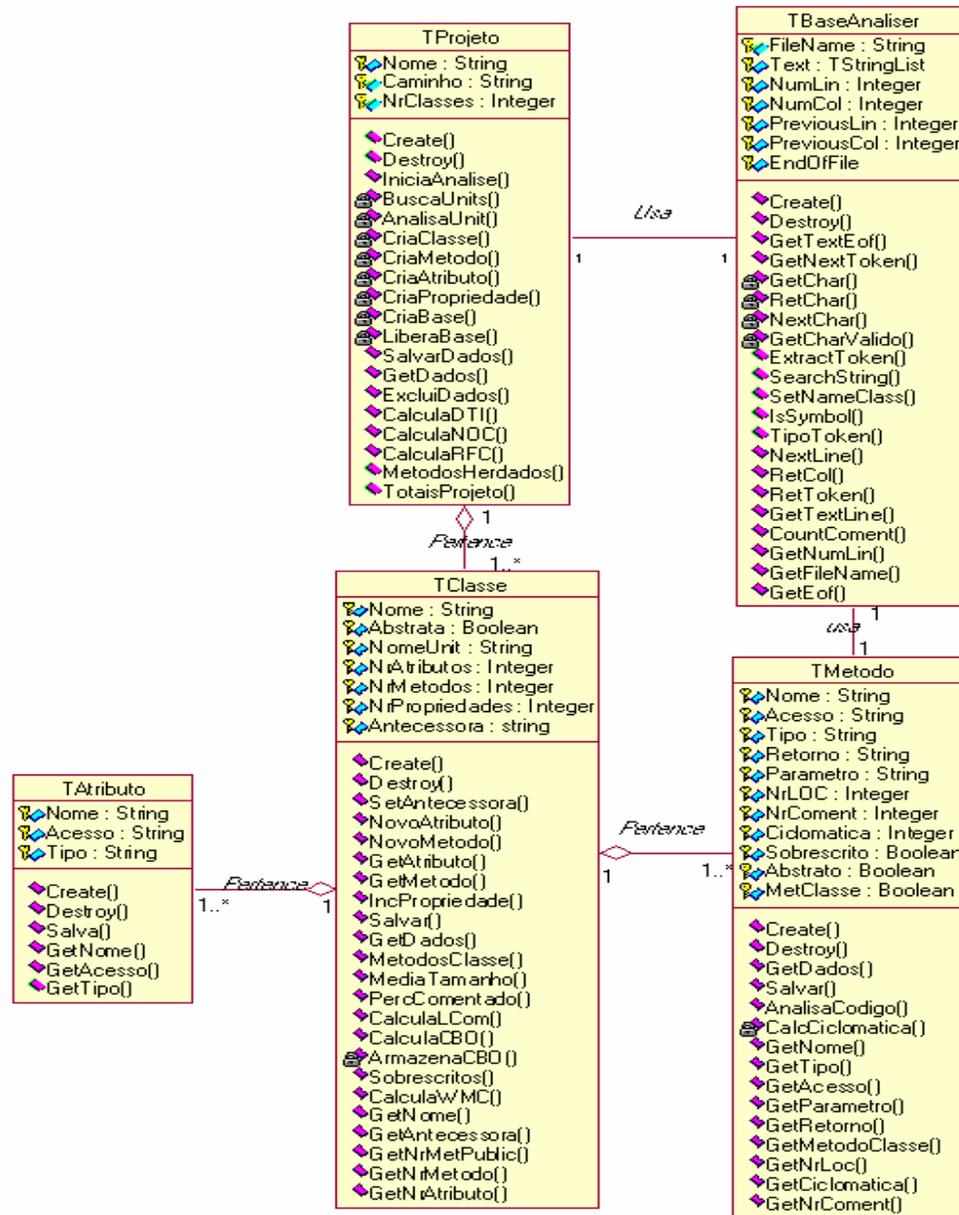
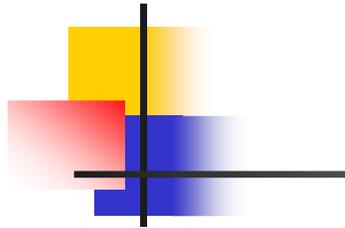
- Uml – ferramenta Rational Rose

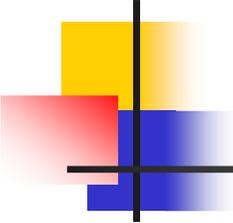
Diagrama de Casos de uso



Especificação Diagrama de classes



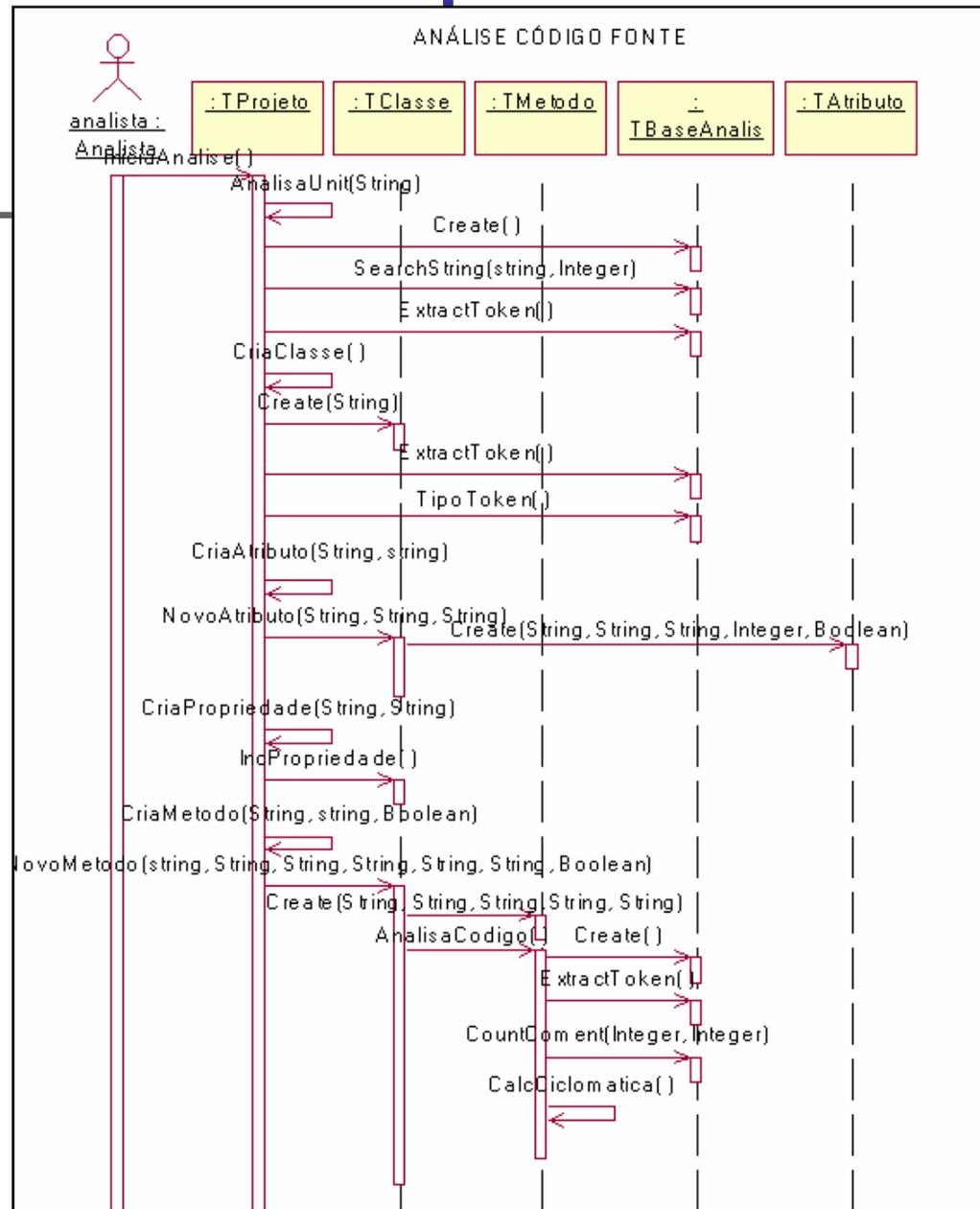
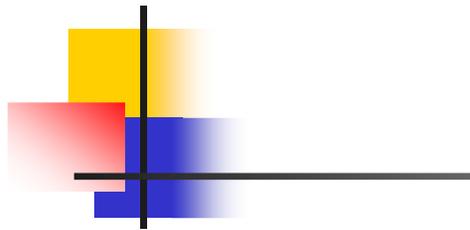


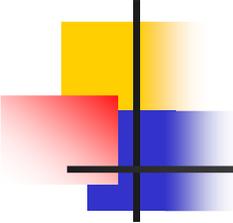


Funções das classes

- TProjeto
 - Analisar *units*
 - Criar TClasse, TAtributo e TMetodo
 - Métricas que precisam de outras classes
- TBaseAnaliser
 - Auxilia análise do código fonte
- TClasse
 - Informações das classes extraídas
- TAtributo
 - Informações dos atributos
- TMetodo
 - Informações dos métodos

Diagrama de seqüência



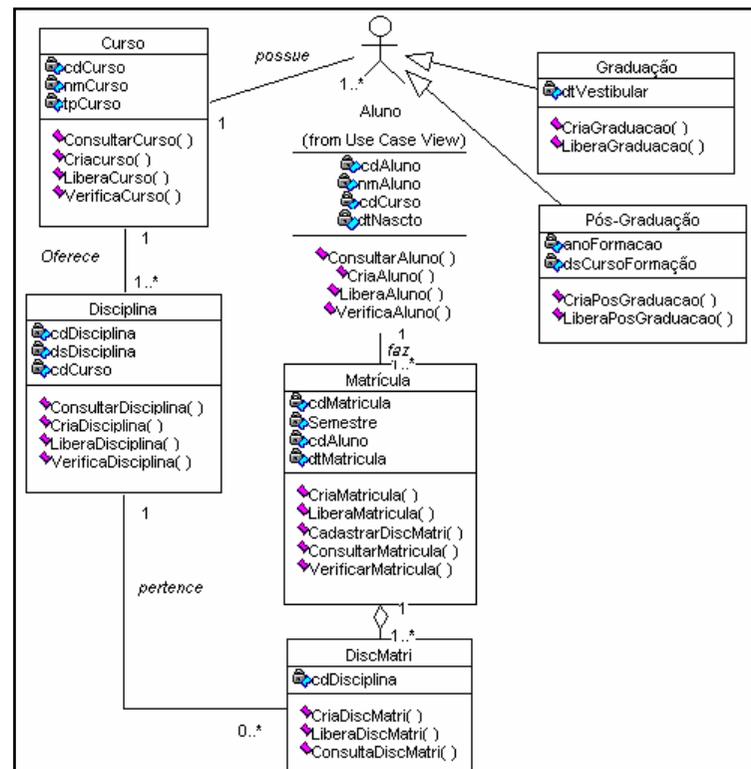


Implementação

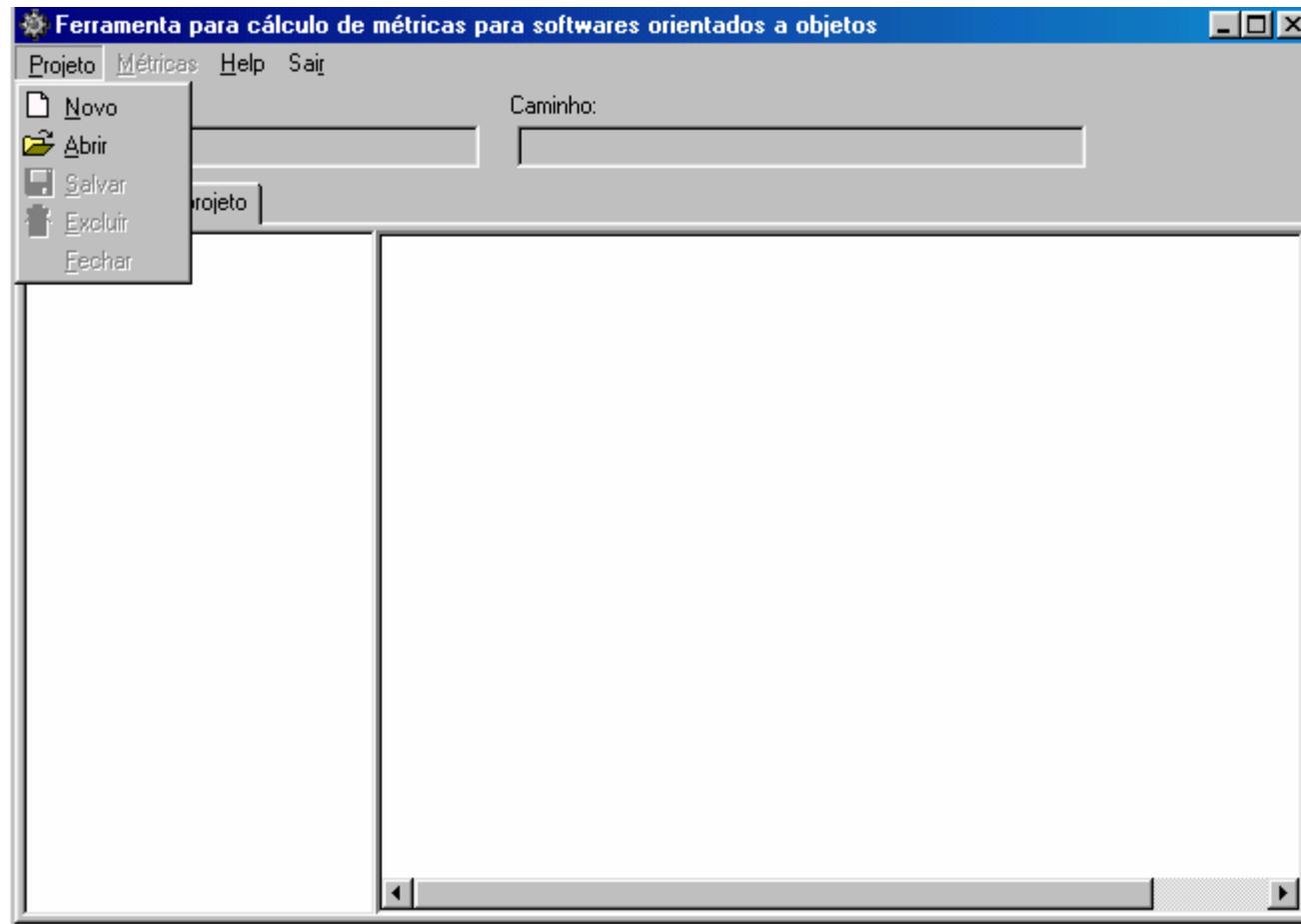
- Técnicas e ferramentas utilizadas
 - Ambiente de desenvolvimento Delphi 5.0
 - Orientação a objetos
- Detalhes da implementação
 - Análise do código fonte
 - TProjeto analisa units
 - TBaseAnaliser extrai Tokens
 - TClasse, TMetodo e Tatributos criadas durante a análise
 - Cálculo das métricas
 - Contagem dos atributos das classes
 - Percorrer classes do projeto
 - Varrer o código fonte do projeto

Operacionalidade da implementação

- Utilizado o estudo de caso de um sistema de controle de matrícula

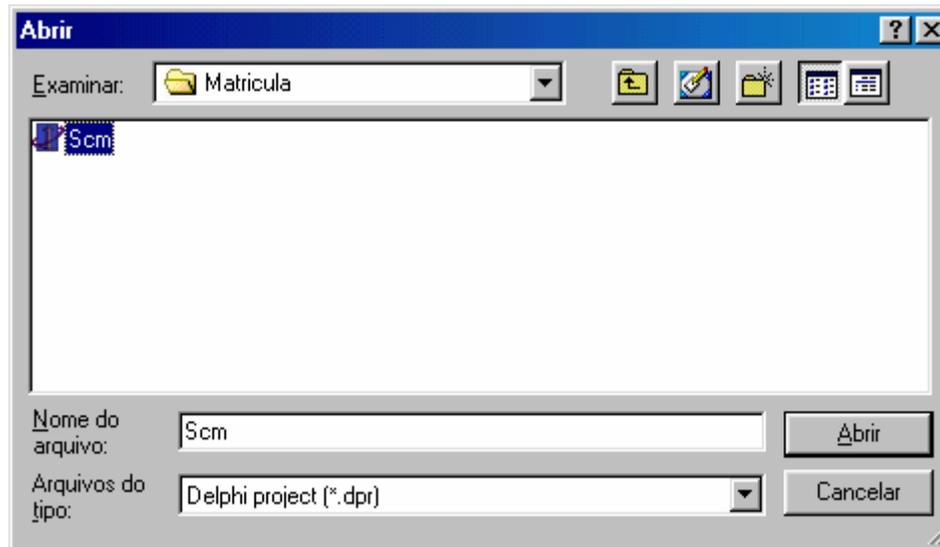


Tela inicial do protótipo

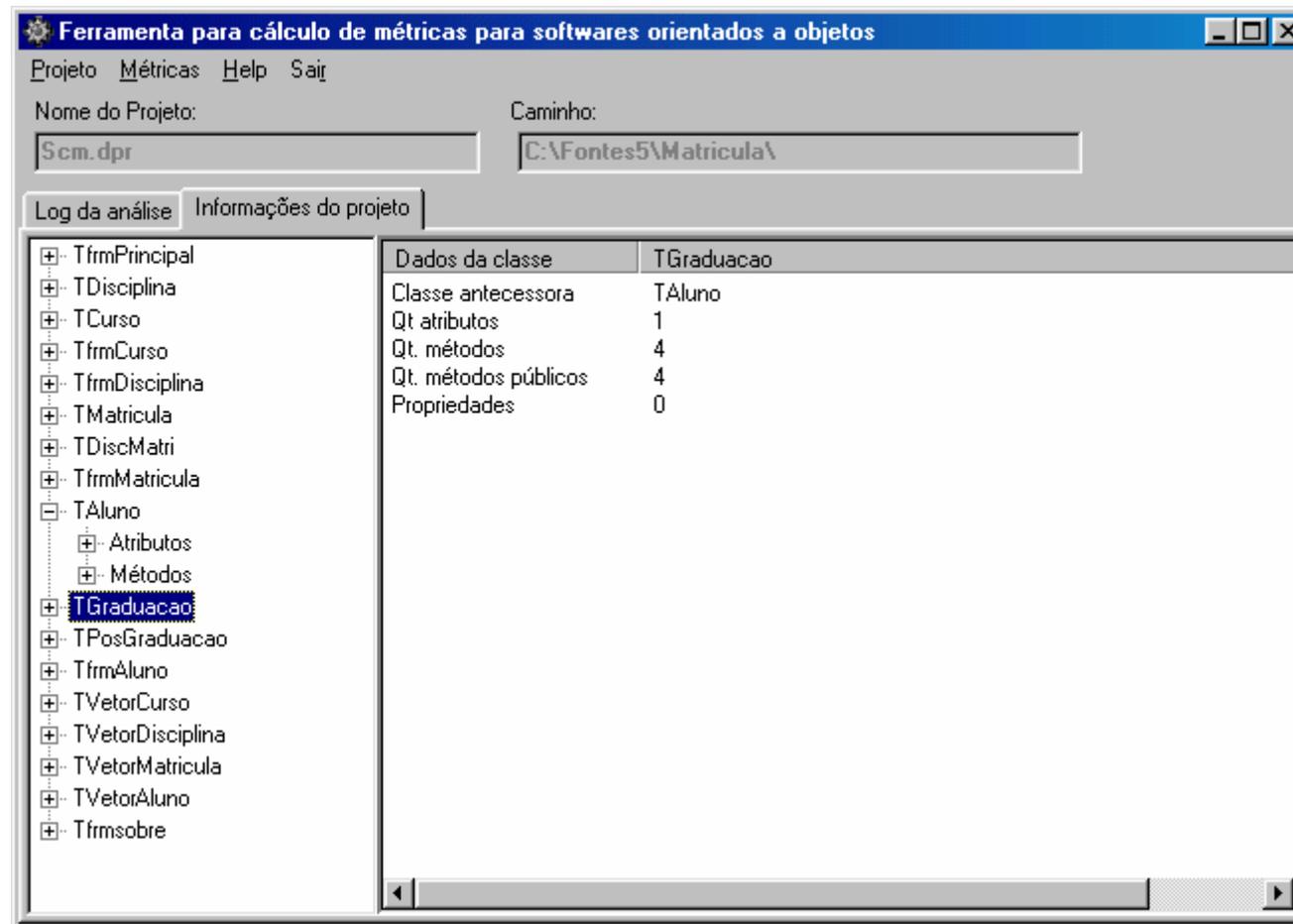


Tela de abertura do projeto

- Abre arquivo com extensão dpr
- Executa a análise do código fonte



Tela informações do projeto



Ferramenta para cálculo de métricas para softwares orientados a objetos

Projeto Métricas Help Sair

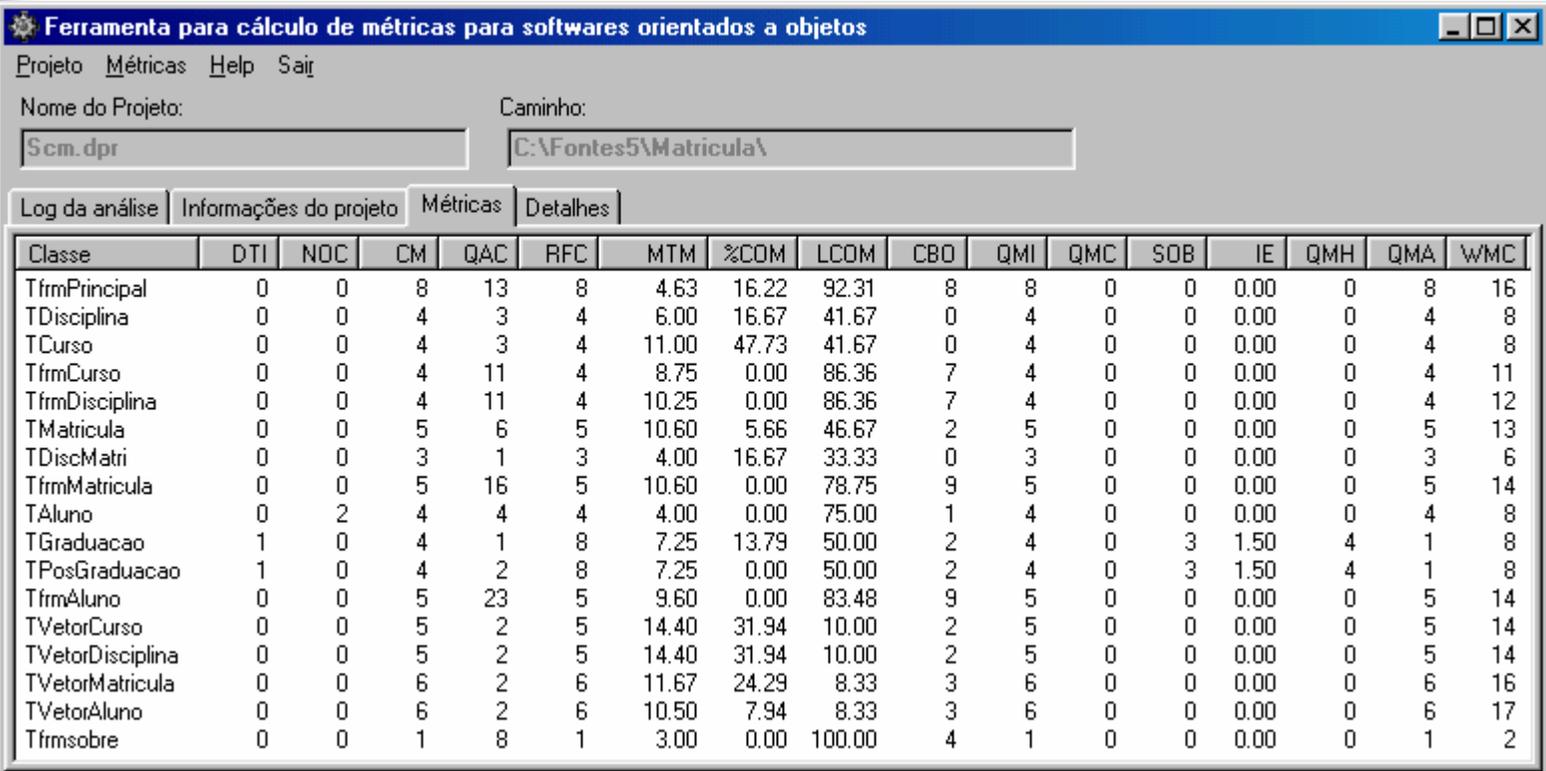
Nome do Projeto: Scm.dpr Caminho: C:\Fontes5\Matricula\

Log da análise Informações do projeto

Dados da classe	TGraduacao
Classe antecessora	TAluno
Qt atributos	1
Qt. métodos	4
Qt. métodos públicos	4
Propriedades	0

- + TfirmPrincipal
- + TDisciplina
- + TCurso
- + TfirmCurso
- + TfirmDisciplina
- + TMatricula
- + TDiscMatri
- + TfirmMatricula
- TAluno
 - + Atributos
 - + Métodos
- + **TGraduacao**
- + TPosGraduacao
- + TfirmAluno
- + TVetorCurso
- + TVetorDisciplina
- + TVetorMatricula
- + TVetorAluno
- + Tfirm sobre

Tela de resultados das métricas



Ferramenta para cálculo de métricas para softwares orientados a objetos

Projeto Métricas Help Sair

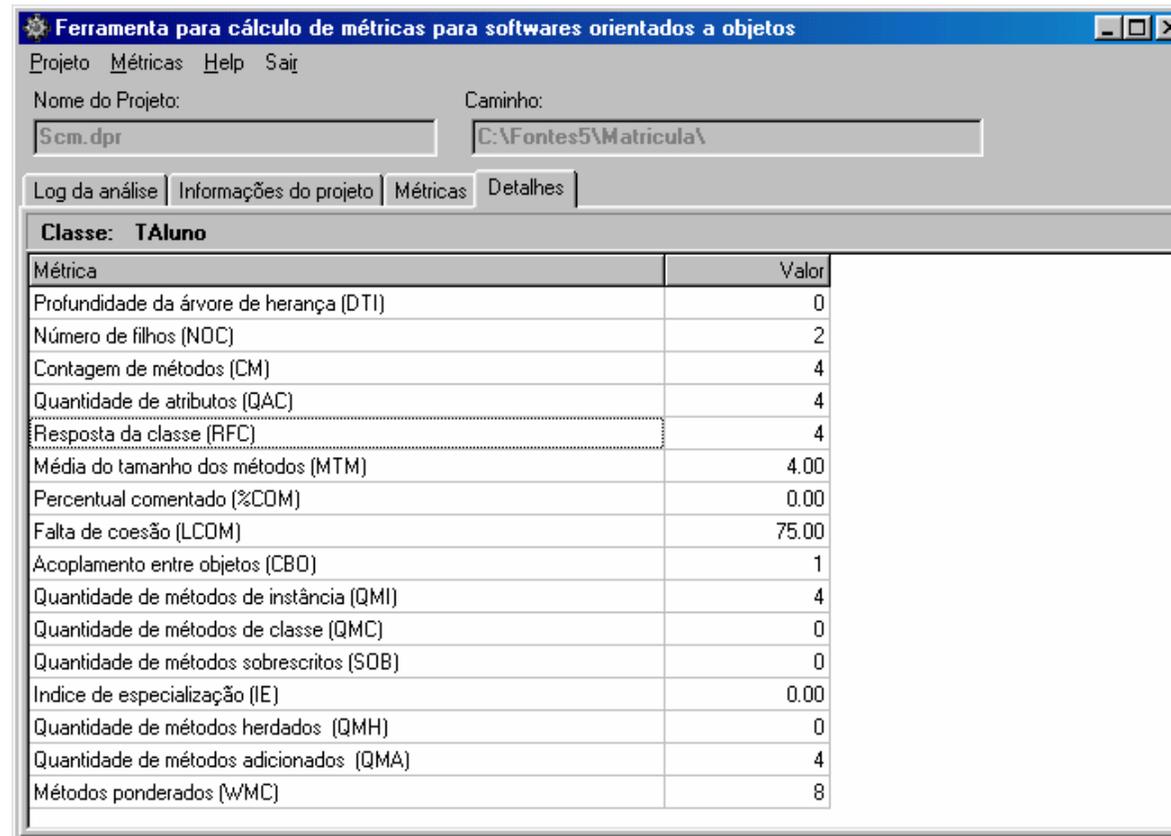
Nome do Projeto: Caminho:

Log da análise | Informações do projeto | **Métricas** | Detalhes

Classe	DTI	NOC	CM	QAC	RFC	MTM	%COM	LCOM	CBO	QMI	QMC	SOB	IE	QMH	QMA	WMC
TfrmPrincipal	0	0	8	13	8	4.63	16.22	92.31	8	8	0	0	0.00	0	8	16
TDisciplina	0	0	4	3	4	6.00	16.67	41.67	0	4	0	0	0.00	0	4	8
TCurso	0	0	4	3	4	11.00	47.73	41.67	0	4	0	0	0.00	0	4	8
TfrmCurso	0	0	4	11	4	8.75	0.00	86.36	7	4	0	0	0.00	0	4	11
TfrmDisciplina	0	0	4	11	4	10.25	0.00	86.36	7	4	0	0	0.00	0	4	12
TMatricula	0	0	5	6	5	10.60	5.66	46.67	2	5	0	0	0.00	0	5	13
TDiscMatri	0	0	3	1	3	4.00	16.67	33.33	0	3	0	0	0.00	0	3	6
TfrmMatricula	0	0	5	16	5	10.60	0.00	78.75	9	5	0	0	0.00	0	5	14
TAluno	0	2	4	4	4	4.00	0.00	75.00	1	4	0	0	0.00	0	4	8
TGraduacao	1	0	4	1	8	7.25	13.79	50.00	2	4	0	3	1.50	4	1	8
TPosGraduacao	1	0	4	2	8	7.25	0.00	50.00	2	4	0	3	1.50	4	1	8
TfrmAluno	0	0	5	23	5	9.60	0.00	83.48	9	5	0	0	0.00	0	5	14
TVetorCurso	0	0	5	2	5	14.40	31.94	10.00	2	5	0	0	0.00	0	5	14
TVetorDisciplina	0	0	5	2	5	14.40	31.94	10.00	2	5	0	0	0.00	0	5	14
TVetorMatricula	0	0	6	2	6	11.67	24.29	8.33	3	6	0	0	0.00	0	6	16
TVetorAluno	0	0	6	2	6	10.50	7.94	8.33	3	6	0	0	0.00	0	6	17
Tfrmsobre	0	0	1	8	1	3.00	0.00	100.00	4	1	0	0	0.00	0	1	2

Tela de detalhes das métricas

- Detalhes das métricas por classe



Ferramenta para cálculo de métricas para softwares orientados a objetos

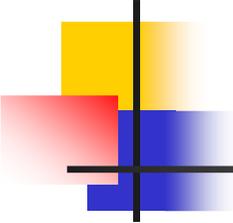
Projeto Métricas Help Sair

Nome do Projeto: Caminho:

Log da análise | Informações do projeto | Métricas | **Detalhes**

Classe: TAluno

Métrica	Valor
Profundidade da árvore de herança (DTI)	0
Número de filhos (NOC)	2
Contagem de métodos (CM)	4
Quantidade de atributos (QAC)	4
Resposta da classe (RFC)	4
Média do tamanho dos métodos (MTM)	4.00
Percentual comentado (%COM)	0.00
Falta de coesão (LCOM)	75.00
Acoplamento entre objetos (CBO)	1
Quantidade de métodos de instância (QMI)	4
Quantidade de métodos de classe (QMC)	0
Quantidade de métodos sobrescritos (SOB)	0
Índice de especialização (IE)	0.00
Quantidade de métodos herdados (QMH)	0
Quantidade de métodos adicionados (QMA)	4
Métodos ponderados (WMC)	8



Conclusões

- A orientação a objetos surge como uma possibilidade para melhoria da qualidade e produtividade do software.
- O uso de medidas, de uma forma organizada e projetada, possui efeito benéfico.
- O protótipo desenvolvido pode auxiliar os gerentes e desenvolvedores a avaliar a qualidade e produtividade do software.

Conclusões

Limitações do protótipo

- Realiza a análise somente de código fonte implementado em Delphi
- Analisa as units especificadas no projeto selecionado (arquivo .dpr)