

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Guia para desenvolvimento do conteúdo do Trabalho de Conclusão de Curso – TCC

1 Introdução

Este guia tem por objetivo, orientar aos alunos e professores na elaboração e condução do Trabalho de Conclusão de Curso - TCC, bem como detalhar a abordagem e procedimentos utilizados no Curso de Tecnologia em Análise e Desenvolvimento de Sistemas. O ponto central para o TCC está na aplicação prática dos conceitos e assuntos envolvendo **ANALISE E DESENVOLVIMENTO DE SISTEMAS**.

1.1 Apresentação Oral e Escrita

O TCC é dividido em duas partes: o trabalho escrito (TE) e a apresentação e defesa (AD) desse trabalho. O trabalho escrito deverá ser formatado de acordo com o padrão adotado pela Instituição e disponível na biblioteca. Deverá ser entregue na secretaria em três vias, encadernadas em espiral na data programada e divulgada pelo professor da disciplina de Gestão de Projetos. O grupo será avaliado de acordo com os seguintes itens:

- Pontualidade na entrega
- Conteúdo apresentado
- Organização do trabalho

A apresentação e defesa deverá ser realizada pelo grupo à banca examinadora em data programada e divulgada pela coordenação. Serão avaliados os seguintes itens:

- Clareza na apresentação – nota individual
- Organização da apresentação – nota do grupo
- Corretude das respostas à banca – nota individual
- Domínio do Conhecimento – nota individual

Para a apresentação Oral o grupo deverá elaborar slides contendo os principais tópicos do trabalho. O tempo para apresentação é de 30 minutos. Todos os componentes do grupo deverão participar da apresentação, o componente que não “falar” ou não comparecer receberá nota zero nos itens AD.

1.2 Sistema de Avaliação

O aluno será considerado aprovado se obtiver média igual ou superior a 7,0. A média do TCC é obtida calculando-se a média simples entre os itens da AD e do TE. O grupo somente será considerado aprovado após a entrega do volume encadernado em capa dura à secretaria. Este volume será disponibilizado na biblioteca.

1.3 Desenvolvimento e Orientação do TCC

O desenvolvimento do TCC deverá ser realizado na disciplina Gestão de Projetos. O professor de Gestão de Projetos é o responsável por estabelecer o cronograma de trabalho e os respectivos prazos para apresentação e conclusão do TCC. Cabe a ele orientar o desenvolvimento do trabalho, no durante as aulas de Gestão de Projetos.

Os grupos também podem ter um professor orientador para as atividades desenvolvidas. Cabe a esse professor realizar os encontros com o grupo para avaliar tecnicamente o conteúdo desenvolvido. O grupo é responsável por contar e convidar um dos professores da Instituição para o essa tarefa e estabelecer com ele uma rotina de trabalho.

1.4 Conteúdo do Trabalho de Conclusão de Curso

Este item tem como objetivo nortear o trabalho dos alunos e professores no sentido de termos um processo de desenvolvimento e avaliação sistematizado e uniforme. A seguir são listados os itens que devem compor o conteúdo do trabalho impresso, cabe destacar que deverá ser seguido o padrão de formatação recomendado pela Faculdade IBTA, disponível na biblioteca, no qual são descritos os itens correspondentes ao pré-texto e pós-texto, tais como capa, dedicatória, introdução, conclusão, referências bibliográficas, glossário, entre outros. Cabe destacar que a introdução deverá descrever a motivação para o desenvolvimento do trabalho, os objetivos a serem alcançados no trabalho, a Justificativa para a escolha do tema, as metodologias utilizadas (pesquisa de campo, utilização de laboratório para testes, pesquisa teórica, etc) e a organização do trabalho (só a apresentação dos capítulos).

1.4.1 Sumário Técnico

A seguir são listados os tópicos que deverão ser contemplados no trabalho.

- **Levantamento de Dados e Análise de Requisitos**
 - Documento de Visão
 - Especificação Suplementar
 - Descrição do Negócio
 - Representação gráfica do Negócio (BPM)
 - Delimitação do Escopo do Sistema
 - Relatório Sintético de Caso de Uso
 - Modelo de Caso de Uso - Especificação UC's
 - Descrição detalhada dos Casos de Uso
 - Diagrama de Classes de Domínio e/ou Modelo de Dados (MER – Modelo Entidade-Relacionamento)
 - Protótipo de Baixa Fidelidade
- **Gestão do Projeto**
 - SOW
 - Project Charter*
 - Registro dos *Stakeholder*
 - Plano de Gerenciamento do Projeto
 - Declaração do Escopo
 - WBS
 - Cronograma
 - Riscos do Projeto
- **Design do Software**
 - Protótipo das Telas - Protótipo de Alta Fidelidade
 - Projeto do Banco de Dados
 - Modelo Físico de Dados

- Dicionário de Dados
- Matriz de Rastreabilidade (CRUD)
- Realização de Casos de Uso
- Diagrama de Classe de Implementação
- Diagrama de Sequência
- Especificação dos Métodos
- Diagrama de Atividades
- Diagrama de Componentes
- Diagrama de Implantação

- **Implementação**

- **Testes**

- Casos de Teste
- Plano de Teste
- Resultados dos Testes

Nos capítulos seguintes serão apresentados modelos e orientações para a elaboração do conteúdo listado no sumário técnico.

2 Levantamento de Dados e Análise de Requisitos

2.1 Documento de Visão

A visão que o usuário ou o cliente tem do produto a ser desenvolvido, especificada no nível das principais necessidades dos envolvidos e recursos do sistema. A referência para confecção do documento de visão é o RUP, disponível em <http://www.wthreex.com/rup>.

2.1.1 Elementos que devem ser contemplados no documento de visão:

- Descrição do Problema: Deve-se descrever o problema que está sendo resolvido com o projeto.
- Descrições dos Envolvidos e dos Usuários
 - Resumo dos Envolvidos. Deve-se descrever os cargos e detalhar as responsabilidades.
 - Resumo dos Usuários. Deve-se descrever os papéis e responsabilidades.
 - Resumo das Principais Necessidades dos Envolvidos ou dos Usuários.
- Visão Geral do Sistema Proposto
 - Objetivos do sistema
 - Descrição do Escopo do projeto
 - Características do sistema
 - Impactos gerados pelo sistema
- Requisitos do Sistema

2.2 Especificação Suplementar

As Especificações Suplementares capturam os requisitos de sistema que não foram identificados imediatamente na modelagem de casos de uso, de acordo com RUP¹ entre os requisitos estão incluídos:

- Requisitos legais e de regulamentação e padrões de aplicativo.
- Atributos de qualidade do sistema a ser criado, incluindo requisitos de usabilidade, confiabilidade, desempenho e suportabilidade.
- Outros requisitos, como sistemas operacionais e ambientes, requisitos de compatibilidade e restrições de design.

A seguir estão listados os elementos que devem ser contemplados na Especificação Suplementar. A descrição de cada item é fiel a descrição do RUP, cuja especificação completa pode ser consultada no site¹.

a. Introdução

A **Especificação Suplementar** captura os requisitos de sistema que não são capturados imediatamente nos casos de uso do modelo de casos de uso. Entre os requisitos estão incluídos:

- Requisitos legais e reguladores, incluindo padrões de aplicativo.
- Atributos de qualidade do sistema a ser criado, incluindo requisitos de usabilidade, confiabilidade, desempenho e suportabilidade.
- Outros requisitos, como sistemas operacionais e ambientes, requisitos de compatibilidade e restrições de design.

I. Funcionalidade

Esta seção descreve os requisitos funcionais do sistema que são expressos no estilo de linguagem natural.

II. Confiabilidade

Os requisitos de confiabilidade do sistema devem ser especificados aqui. Abaixo, algumas sugestões:

- Disponibilidade - especifique a porcentagem de tempo disponível (xx.xx%), as horas de uso, o acesso à manutenção, as operações de modo degradado etc.
- Tempo Médio entre Falhas (MTBF) - normalmente especificado em horas, mas também poderá ser especificado em termos de dias, meses ou anos.
- Tempo Médio para Reparo (MTTR) - quanto tempo o sistema poderá ficar sem funcionar após uma falha?
- Exatidão - especifique a precisão (resolução) e exatidão (através de algum padrão conhecido) necessárias na saída dos sistemas.
- Taxa máxima de erros ou defeitos - geralmente expressa em termos de erros/KLOC (*thousands of lines of code*) ou de erros/ponto de função.
- Taxa de erros ou defeitos - categorizados em termos de erros pouco importantes, importantes e críticos: o(s) requisito(s) devem definir o que se entende por um erro "crítico" (ex: perda total de dados ou total incapacidade de usar determinadas partes da funcionalidade do sistema).

¹ <http://www.wthreex.com/rup/>

III. Suportabilidade

Esta seção indica todos os requisitos que aprimorarão a suportabilidade ou manutenibilidade do sistema que está sendo criado, incluindo padrões de codificação, convenções de nomeação, bibliotecas de classes, acesso à manutenção e utilitários de manutenção.

2.3 Relatório Sintético de Caso de Uso

De acordo com RUP (2008) a descrição do relatório sintético do modelo de casos de uso deve:

- Estabelecer quais são os principais casos de uso do sistema (o motivo pelo qual o sistema foi construído).
- Resumir os fatos técnicos importantes sobre o sistema.
- Indicar as delimitações do sistema; o que o sistema não deve executar.
- Resumir o ambiente do sistema como, por exemplo, plataformas-alvo e software existente.
- Descrever qualquer sequência em que os casos de uso são executados normalmente no sistema.
- Especificar a funcionalidade não tratada pelo modelo de casos de uso.

O modelo padrão para descrição do relatório sintético dos casos de uso é apresentado e exemplificado a seguir:

Tabela 1 – Modelo de Relatório Sintético de Casos de Uso

Nome do Caso de Uso	Descrição
UC01 – Cadastrar Pedido	

2.4 Diagrama de Casos de Uso

O diagrama de Casos de Uso tem o objetivo de auxiliar a comunicação entre os analistas e o cliente, estabelece um contrato entre as partes sobre as funcionalidades que deverão ser contempladas pelo sistema. O diagrama representado os Atores, os Casos de Uso e os Relacionamentos. A seguir esses elementos são brevemente explicados e a notação adotada para sua representação é exemplificada.

- **Atores do sistema:** Entidade que interage com o sistema, pode ser representada por papéis executados por usuário, por exemplo, Cliente, hardware externo ou sistemas externos.



Figura 1 – Representação UML do Ator do Caso de Uso

- **Casos de Uso:** Representa uma interação entre o Ator e o Sistema; compreende os requisitos funcionais do sistema.

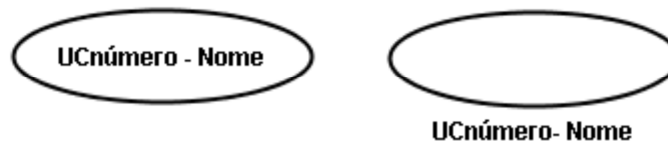


Figura 2 - Representação UML do Caso de Uso

O Caso de Uso deve ser identificado pela sigla UC (Use Case) e o número atribuído ao Caso de Uso seguido do seu nome, por exemplo, UC03 – Fazer Pedido. A identificação do Caso de Uso pode ser colocada dentro da elipse ou logo abaixo dela.

- Associações entre Atores e Casos de Uso:



Figura 3 - Associação entre ator e caso de uso

- Generalizações entre os Atores:

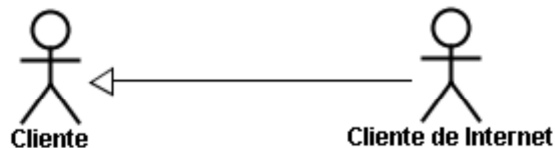


Figura 4 - Generalização entre Atores

Na figura 4 o Cliente de Internet é uma generalização do ator Cliente, isto é, é um tipo de cliente.

- Generalizações entre os Casos de Uso.

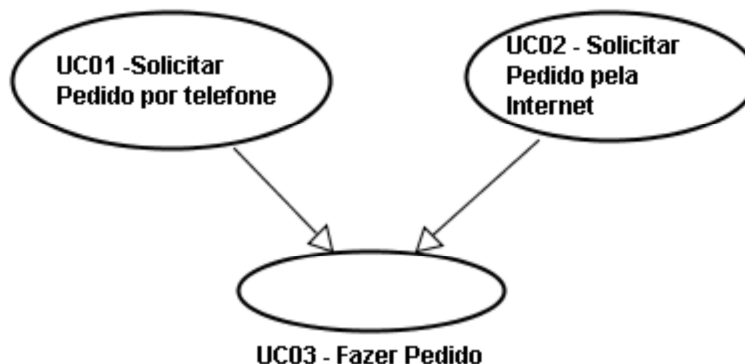


Figura 5 - Generalizações entre Casos de Uso

- Extensões (*extend*) entre os Casos de Uso:

As extensões de Caso de Uso indicam que o Caso de Uso para onde a seta aponta poderá acionar a funcionalidade do Caso de Uso relacionado de acordo com uma determinada condição.

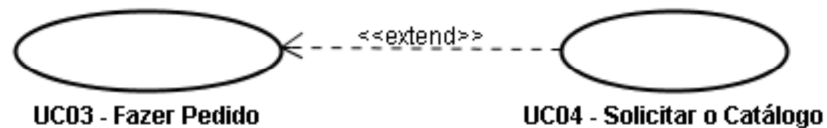


Figura 6 - *Extends* entre casos de uso

- Inclusões (*include*) entre os Casos de Uso:

As inclusões indicam que o Caso de Uso que aponta inclui a funcionalidade do Caso de Uso apontado à sua, isto é, sempre que é acionado ele aciona o comportamento do Caso de Uso apontado. No exemplo da figura 7, quando o Caso de Uso UC03 – Fazer pedido é acionado ele executa o Caso de Uso UC05 – Fornecer os Dados do Cliente.

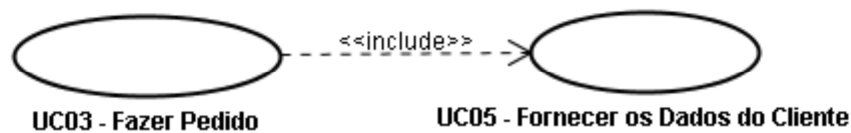


Figura 7 - *Includes* em casos de uso

A figura 8 ilustra a relação entre Atores e Casos de Uso apresentados nos exemplos anteriores.

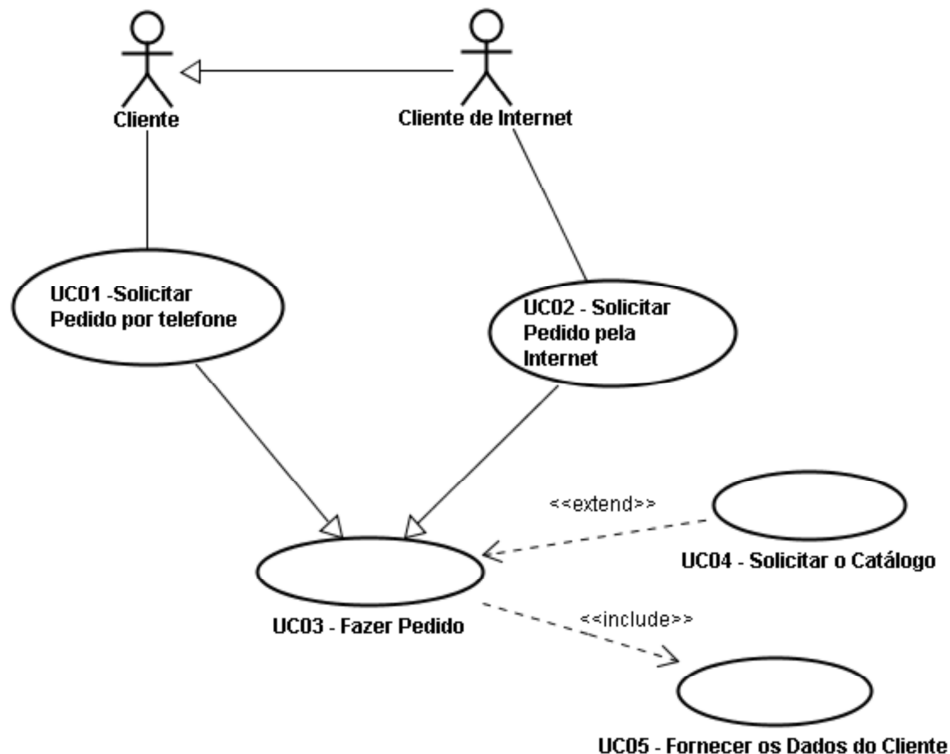


Figura 8 - Diagrama de casos de uso (adaptado de RUP, 2008)

2.5 Descrição dos Casos de Uso

A descrição dos Casos de Uso esclarece a lógica para execução da atividade desempenhada pelo Caso de Uso. Na tabela apresentada a seguir, é apresentado um modelo para descrição dos Casos de Uso, em http://www.wthreeex.com/rup/process/modguide/md_uc.htm#Are%20All%20Use%20Cases%20Described%20in%20Detail? estão disponíveis explicações e exemplos adicionais sobre os itens da tabela.

Tabela 2 - Modelo para descrição dos Casos de Uso

Nome do Caso de Uso:	Nome do Caso de Uso exemplo UC01- Preencher Cadastro
Resumo:	Descrição sintética do caso de uso
Ator Principal:	Ator que aciona o caso de uso
Pré-condição	Ação, atividade ou estado que deve ocorrer antes do Caso de Uso ser acionado.
Pós-condição	Ação, atividade ou estado gerado após a execução do caso de uso.
Fluxo Principal: Descrição da lógica de execução do Caso de Uso (seu algoritmo). Diretrizes para identificação do fluxo: FP01, FP02, FPNN	
Fluxo Alternativo: Descrição das ações que deverão ser realizadas quando ocorrer alguma ação inesperada no fluxo principal. Diretrizes para identificação do fluxo alternativo: FA01, FA02, FANN	

2.6 Diagrama de Classes de Domínio

O diagrama de Classes de Domínio é correspondente ao modelo conceitual, nesse item deve-se representar os elementos relevantes para o negócio ou problema, tais como Funções desempenhadas por pessoas, coisas tangíveis, interações, entre outros. No exemplo a seguir podemos visualizar um modelo de classes e na figura 10 as representações para associações de Classes.

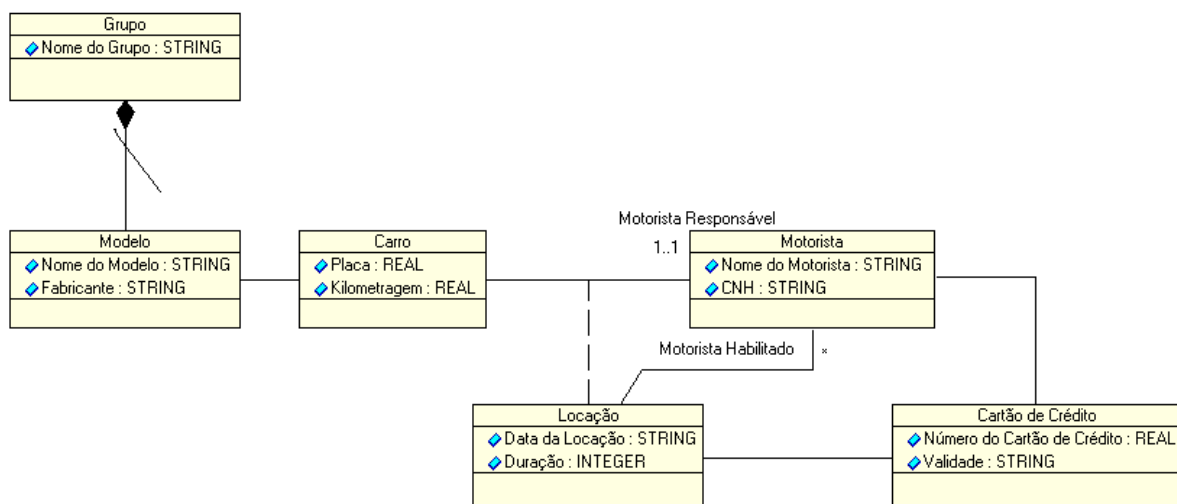


Figura 9 - Exemplo de Diagrama de Classes de Domínio

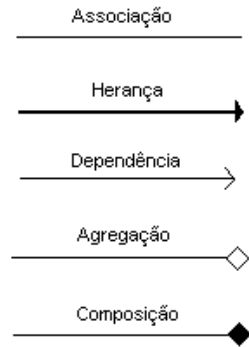


Figura 9 - Representação das relações entre classes

3 Gestão do Projeto

Compete ao grupo escolher dentre os integrantes aquele que exercerá a função de Gerente do Projeto. O gerenciamento do Projeto deverá seguir os métodos, técnicas e ferramentas vistos na disciplina Gerenciamento de Projetos de Software.

4 Projeto do Software

4.1 Protótipo das Telas

Deverá ser desenvolvido e documentado o protótipo das telas, considerando-se os aspectos de ergonomia e usabilidade. O recurso utilizado para o desenvolvimento dos protótipos é de livre escolha do grupo, entretanto as interfaces devem representar a realidade do sistema proposto, com suas devidas funcionalidades.

4.2 Projeto do Banco de Dados

4.2.1 Modelo Conceitual de Dados

De acordo com Cougo (1997, p. 28) é aquele em que os objetos, suas características e relacionamentos têm a representação fiel ao ambiente observado, independente de limitações quaisquer impostas por tecnologias, técnicas de implementação ou dispositivos físicos.



Figura 10 - Modelo Conceitual de Dados – (Parte extraída do Modelo IsoManager de Cândido et al (2007))

4.3 Modelo Físico de Dados

De acordo com Cougo (1997, p. 30) é aquele em que a representação dos objetos é feita sob o foco do nível físico de implementação das ocorrências, ou instâncias das entidade e seus relacionamentos. Nessa etapa deve-se definir qual será o sistema gerenciador do banco de dados e estruturar os campos e relacionamentos de maneira compatível com ele.

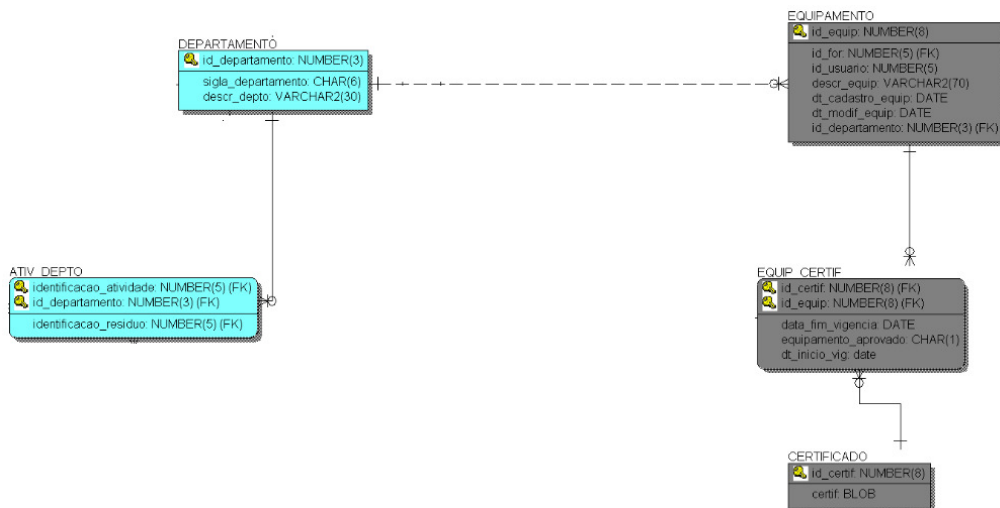


Figura 11 - Modelo Físico de Dados – (Parte extraída do Modelo IsoManager de Cândido et al 2007))

4.4 Dicionário de Dados

De acordo com Pressman (2002), o dicionário de dados é uma listagem organizada de todos os elementos de dados que são pertinentes ao sistema, com definições precisas e rigorosas, de forma que tanto o usuário como o analista de sistemas, tenham uma compreensão comum das entradas, das saídas, dos componentes dos depósitos de dados e [até mesmo] dos cálculos intermediários. Na tabela 2 é apresentado um modelo padrão para o dicionário de dados, deverá ser elaborada uma tabela para cada entidade do modelo físico de dados.

Tabela 3 - Dicionário de Dados

Entidade	Nome da Tabela, ex.: Produto				
Descrição	Descrição do que a tabela representa, ex.: Armazenará dados sobre os produtos de uma papelaria				
Exemplo	exemplo de um item que está sendo representado, ex.: Caneta				
Volume esperado	500 registros por ano				
Tempo de retenção	2 anos				
Rotina de limpeza	a cada dois anos os registros são automaticamente apagados do sistema.				
Definição dos Atributos					
Nome	Tipo Dado	Tamanho	Nulidade	Tipo Chave	Descrição

nome do campo		quantidade de caracteres suportado	sim ou não	PK ou FK ou UK	breve descrição do campo
---------------	--	------------------------------------	------------	----------------	--------------------------

4.5 Matriz de Rastreabilidade (CRUD)

CRUD é o acrônimo para *Create, Retrive, Update and Delete*, é utilizada para mapear as operações de inserção, consulta, alteração e exclusão de dados em bancos de dados relacionais ou em interfaces para usuários. Deverão ser apresentadas as matrizes de rastreabilidade para:

- Entidade de Dados X Funcionalidade
- Tabela X *Stored Procedure* (*Procedure, Function e Trigger*)
- Tabela X Visão (*views*)

Para isso criar uma tabela onde deverão ser listadas as entidades e os identificadores que serão utilizados na matriz CRUD:

Tabela 4 - Identificação das Entidades

Nome da Entidade	Identificador
PRODUTO	EN01
CLIENTE	EN02

Em seguida deverá ser apresentada a matriz de rastreabilidade contemplando todas as Entidades, Casos de Uso e *Stored Procedure*, conforme o modelo apresentado na tabela 5:

Tabela 5 - Modelo de Matriz CRUD

Entidades	IDENTIFICADOR DA ENTIDADE			
Casos de uso	C	R	U	D
Identificador do UC				

Na tabela 6 é apresentado um exemplo de preenchimento da matriz CRUD:

Tabela 6 - Exemplo Matriz CRUD

Entidades	EN01				EN02			
Casos de uso	C	R	U	D	C	R	U	D
UC01	X	x						x
UC02	x	x	x	x		x		

4.6 Realização de Casos de Uso

4.6.1 Diagrama de Classe de Implementação

Os Diagrama de Classes deverão ser desenvolvidos de acordo com os padrões da UML. Deverá ser apresentado no formato implementação, isto é, deverá conter a representação dos atributos, métodos, associações e multiplicidades.

Notação:

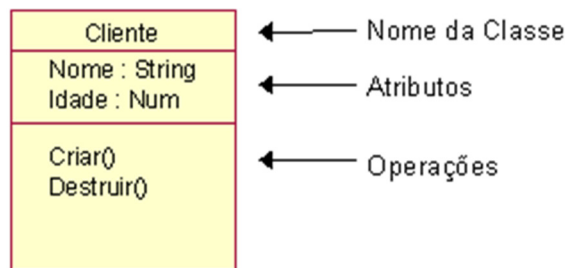


Figura 12 - Representação da Classe

Considerações:

Métodos e atributos estáticos devem aparecer sublinhados
Métodos virtuais aparecem em itálico

Acessibilidade ou Visibilidade dos atributos e métodos:

Protegido
+ Público
- Privado

As associações foram apresentadas na figura 10 do item 2.6 – Classes de Domínio.

4.6.2 Diagrama de Sequência

Os Diagramas de Sequência deverão ser desenvolvidos de acordo com os padrões da UML e definidos em conjunto com o professor da disciplina MOST. Mostra a troca de mensagens no decorrer do tempo entre as Classes na execução de um Caso de Uso.

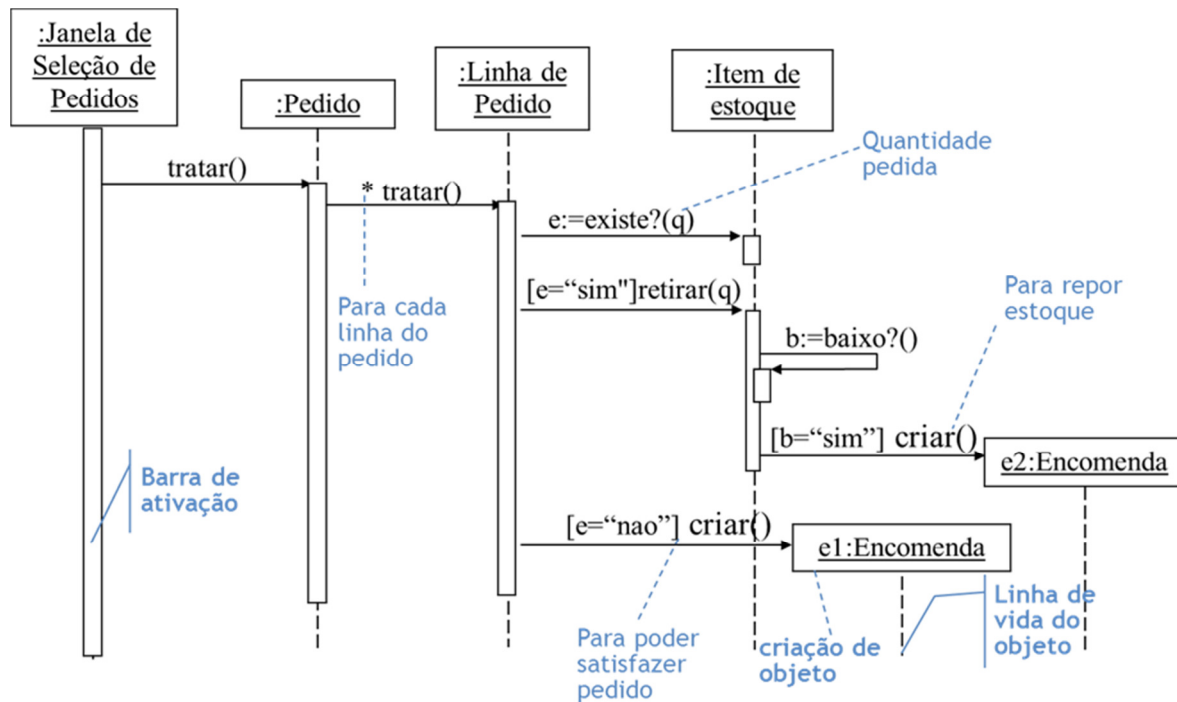


Figura 13 - Diagrama de Sequência - Tratar Pedido de Estoque (Adaptado de Faria, 2001)

Tipos de mensagens:

- síncrona
- ← retorno de mensagem síncrona
- simples
- assíncrona

4.7 Especificação dos Métodos

4.7.1 Diagrama de Atividades

Os Diagramas de Atividades deverão ser desenvolvidos de acordo com os padrões da UML e definidos em conjunto com o professor da disciplina MOST. Sua finalidade é representar o fluxo de execução de um procedimento, processo ou rotina. Detalha os passos de um Caso de Uso.

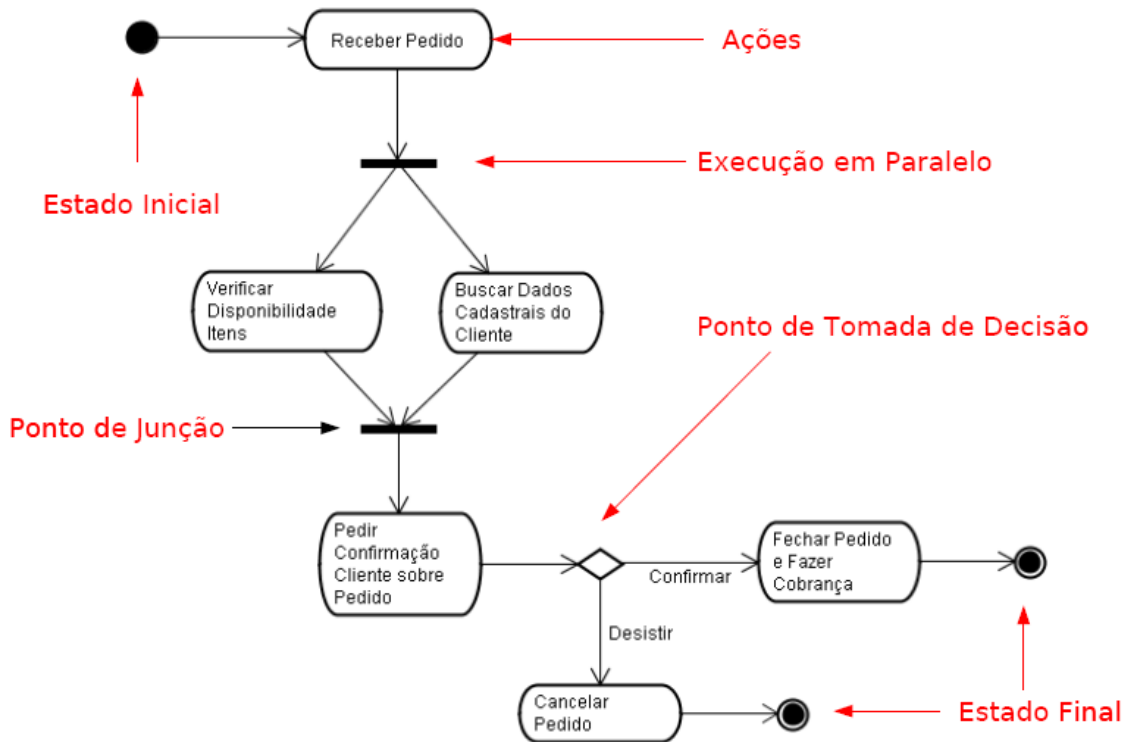


Figura 14 - Diagrama de Atividades (fonte: Queiroz, 2008)

4.7.2 Descritiva

Descrever em Português Estruturado as regras, procedimentos e condições envolvidas no método, cujo conteúdo tenha cerca de uma página (no máximo).

4.8 Diagrama de Componentes

Representa a estrutura de componentes, incluindo os classificadores que eles especificam e os artefatos que implementam.

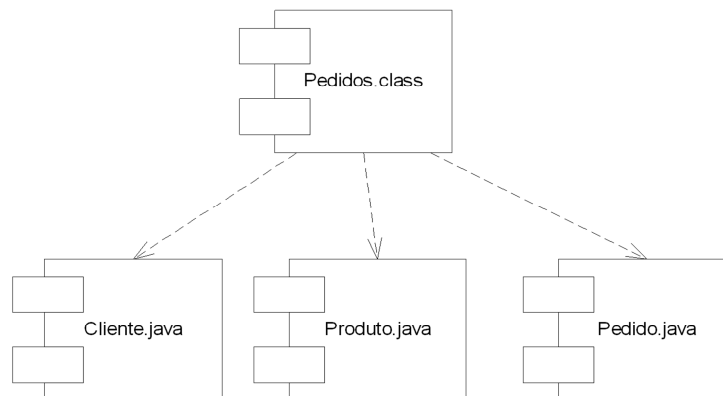


Figura 15 - Exemplo de Diagrama de Componentes

Um componente é uma estrutura física que compõe o sistema, pode ser um arquivo, um documento, uma biblioteca, uma tabela ou outro executável. Na figura 16 são exemplificados os estereótipos que podem ser utilizados.

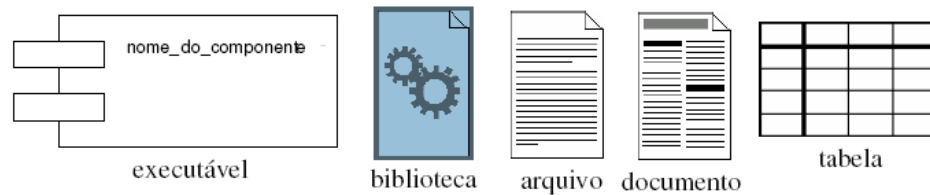
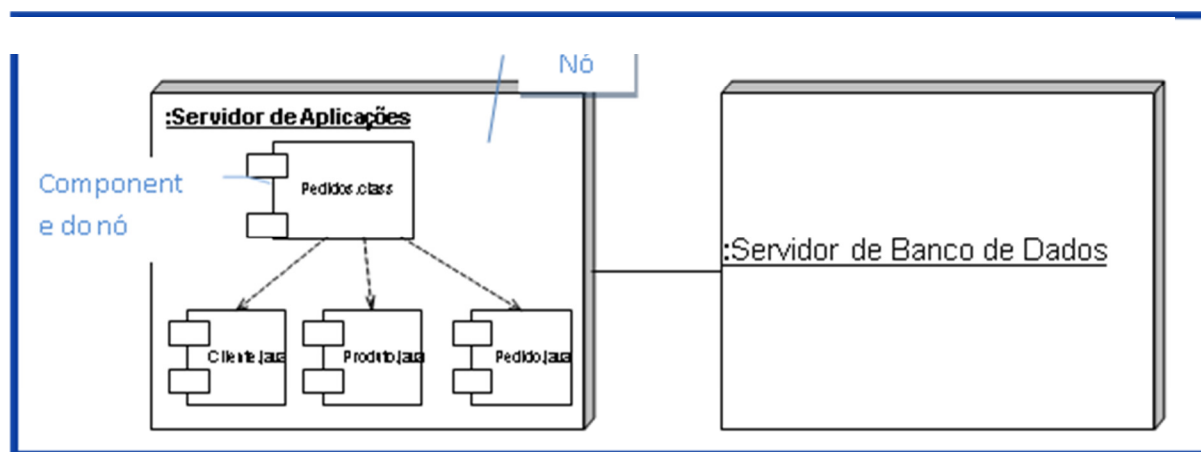


Figura 16 - Estereótipos de componentes

4.9 Diagrama de Implantação ou distribuição

Representa a configuração de elementos de processamento em tempo de execução e os componentes de software, processos e objetos que neles são executados, ilustra a distribuição física do sistema.



4.10 Implementação do Sistema

A implementação do sistema consiste na codificação das rotinas documentadas. O grupo deverá criar o banco de dados. Cada componente do grupo deverá implementar um Caso de Uso e criar uma interface, isto é, deverá codificar e documentar o respectivo programa.

5 Testes

Cada componente do grupo deverá criar um plano de teste, realizar os testes e analisar e documentar os resultados do Caso de Uso e interface por ele codificado. Aspectos que devem ser observados:

- Funcionalidade
- Conformidade com o projeto

Nessa etapa o grupo deverá utilizar o modelo do RUP, apresentado a seguir:

5.1 Plano de Teste

O plano de testes para 1a. iteração da elaboração descreve o planejamento dos testes especificamente para a iteração. Serão abordados os seguintes aspectos:

- os tipos de teste planejados e a cobertura dos requisitos de teste quanto aos casos de uso da iteração 1;
- o ambiente de testes a ser utilizado;
- os recursos humanos e de sistema planejados para as atividades de teste.

a. Prioridades e Contexto das Atividades de Teste

<pacote do caso de uso>

<Caso de Uso 1>

<Caso de Uso 2>

b. Tipos de Teste Considerados

Requisito de Teste	Tipo do Teste	Caso de Uso
<identificador> descrição	<tipo do teste>	<identificador do caso de uso>

c. Cobertura dos testes quanto aos Casos de Uso da iteração

Cobertura do Planejamento de Teste <link matriz de relacionamento entre caso de uso, requisito de teste e caso de teste>

d. Especificação dos Casos de Teste

Caso de Teste	Descrição	Passo a passo	Resultado	Pré-Condição	Requisito
<identificador do caso de teste>: <caso de teste>	<descrição>	<passo a passo>	<resultado>	<pre-condição>	TR1

Referências

CÂNDIDO, A. et. Al. Projeto IsoManager. Trabalho de Conclusão de Curso. Faculdade IBTA. São Paulo. 2007.

COUGO, P. Sérgio. Modelagem conceitual e projeto de bancos de dados. Rio de Janeiro: Campus, 1997.

PMIMG. Tradução livre do PMBOK 2000, V 1.0, disponibilizada através da Internet pelo PMI MG em janeiro de 2002

PRESSMAN, R. S. Engenharia de Software. São Paulo: Pearson, 2005.

RUP. [sl.], [2008]. Disponível em {<http://www.wthreeex.com/rup/>}. Acesso em 06/dez/2008.