



## **PLATAFORMA DE DESENVOLVIMENTO PINHÃO PARANÁ**

### **Orientações para documentar a Fase de Análise**

**Julho – 2006**

Sumário de Informações do Documento		
<b>Tipo do Documento:</b> Definição		
<b>Título do Documento:</b> Orientações para documentar a Fase de Análise		
<b>Estado do Documento:</b> Elaborado		
<b>Responsáveis:</b> Equipe GIC		
<b>Palavras-Chaves:</b> Processo de Desenvolvimento, Caso de Uso, Especificação		
<b>Resumo:</b> A finalidade do documento é orientar a confecção da documentação de sistema exigida na fase de Análise		
<b>Número de páginas:</b> 19		
<b>Software utilizados:</b> Open Office		
Versão	Data	Mudanças
1.0	19/07/06	Elaborado por Danielle Mayer com colaboração da equipe GTI ( Eleni Hiratsuka, Luciana Costa, Karina Curcio, Manoel Leal, Jeferson Rubel, Daniel Schutz, Garten Nack, Francisco Suma, Cláudio Marcelo, Daniel Ferreira ) Revisão por Cléverson Budel / Hélio Piccinatto / Cíntia Evangelista
1.1	04/10/06	Complementando descrição do item <b>Fluxo Principal do Corpo da Especificação.</b>
1.2	14/03/07	Atualização do documento para incluir os campos de formulário hidden, button, link. Enviado por: GTI – DITES – Karina Curcio

# SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>4</b>
<b>2 ORIENTAÇÕES.....</b>	<b>4</b>
2.1 DIAGRAMA DE CASOS DE USO .....	4
2.1.1 <i>Diagrama de Casos de Uso.....</i>	4
2.1.2 <i>Pacotes de Casos de Uso .....</i>	5
2.2 ESPECIFICAÇÃO DOS CASOS DE USO DE ANÁLISE.....	6
2.2.1 <i>Estruturação.....</i>	6
2.2.1.1 <i>Quadro de informações.....</i>	6
2.2.1.2 <i>Corpo da Especificação.....</i>	7
2.2.1.3 <i>Lembretes e Dicas para especificar um Caso de Uso.....</i>	9
2.2.2 <i>Orientação sobre especificação de campos de Formulário.....</i>	11
2.2.2.1 <i>Listbox / Combobox.....</i>	11
2.2.2.2 <i>Textfield.....</i>	12
2.2.2.3 <i>Label.....</i>	12
2.2.2.4 <i>Radio Button.....</i>	12
2.2.2.5 <i>CheckBox.....</i>	13
2.2.2.6 <i>TextArea.....</i>	13
2.2.2.7 <i>Hidden.....</i>	14
2.2.2.8 <i>Button.....</i>	14
2.2.2.9 <i>Link.....</i>	14
2.2.3 <i>Orientação para Validação de Campo Obrigatório.....</i>	14
2.2.4 <i>Orientação sobre especificação de mensagens.....</i>	15
2.2.5 <i>Orientação sobre a especificação de casos de uso de relatórios.....</i>	15
2.3 MAQUETE (IDENTIDADE VISUAL) / PROTÓTIPO DO SISTEMA .....	17
2.3.1 <i>Estruturação de formulários de Endereço.....</i>	17
2.3.2 <i>Revisão.....</i>	18
2.4 HOMOLOGAÇÃO.....	18
2.4.1 <i>Dicas.....</i>	18
<b>3 REFERÊNCIAS.....</b>	<b>19</b>

## 1 INTRODUÇÃO

O objetivo deste documento é orientar a confecção de alguns dos produtos previstos pela fase de análise.

## 2 ORIENTAÇÕES

Para geração de produtos que seguem a definição da UML (Unified Modeling Language) a Celepar adotou como ferramenta oficial o RSM (Rational Software Modeler).

### 2.1 Diagrama de Casos de Uso

#### 2.1.1 Diagrama de Casos de Uso

A modelagem de um diagrama de caso de uso é uma técnica usada para descrever e definir os requisitos funcionais de um sistema. O diagrama é composto por atores e casos de uso. Os atores representam o papel de uma entidade externa no sistema como um usuário, um *hardware* ou outro sistema que interage com o sistema modelado. Os atores comunicam-se com o sistema através dos casos de uso, onde o caso de uso representa uma funcionalidade executada pelo sistema.

Durante a identificação do Caso de Uso pode-se descobrir novos requisitos. Neste caso, é necessário que a lista de requisitos seja atualizada e o impacto no desenvolvimento do sistema seja avaliado. Os requisitos não funcionais também deverão ser incorporados na lista de requisitos.

Para apoiar a definição dos casos de uso, muitas vezes, é necessária a definição de um **glossário** de termos comuns, tanto do domínio do negócio quanto de informática, para que seja possível um melhor entendimento do vocabulário utilizado no projeto.

O nome de cada caso de uso deve representar o seu objetivo, ou seja, deve indicar o que é alcançado pela interação com o ator, desta forma, muitas vezes o nome precisa ser composto

por várias palavras para ser entendido. O nome de cada caso de uso deve ser único para todo o sistema. Por exemplo: Receber Itens de Depósito, Manter Usuários, Emitir Boletim Escolar.

A tarefa de localização dos atores e casos de uso pode ser feita iterativamente, localizando-se primeiramente os atores e, depois, os casos de uso relacionados ou vice-versa. Segue abaixo um check-list cuja finalidade é auxiliar na identificação de atores e casos de uso:

▪ CHECK-LIST
▪ <b>Atores</b>
▪ Quais grupos de usuários necessitam de ajuda do sistema para executar tarefas?
▪ Quais grupos de usuários são necessários para executar as funções básicas mais óbvias do software?
▪ Quais grupos de usuários deverão executar funções secundárias, como manutenção e administração do software?
▪ O sistema interagirá com algum sistema externo de hardware ou software?
▪ Quem está interessado em certo requisito?
▪ Que atores são requeridos pelos casos de uso?
▪ Algum ator desempenha diversos papéis? Diversos atores desempenham o mesmo papel?
▪ <b>Casos de Uso</b>
▪ Segundo o ator, quais são as principais tarefas a serem executadas pelo sistema?
▪ O ator precisa informar o sistema sobre mudanças externas repentinas?
▪ O ator precisa estar informado sobre certas ocorrências no sistema?
▪ O ator irá iniciar ou desligar o sistema?
▪ O software precisa atualizar outro software?
▪ Os requisitos funcionais identificados estão todos representados no software?
▪ O ator vai criar, armazenar, modificar, remover ou ler alguma informação do sistema?
▪ O software fornece o comportamento correto ao negócio?
▪ Que casos de uso oferecem suporte e manutenção ao sistema?
▪ Todos os requisitos funcionais podem ser atendidos pelos casos de uso?

### 2.1.2 Pacotes de Casos de Uso

Se os casos de uso cresceram muito nessa fase, eles devem ser divididos em pacotes de casos

de uso para simplificar a manutenção do modelo de casos de uso. Isso também facilita a compreensão desse modelo e simplifica a atribuição de responsabilidades, permitindo que os pacotes de casos de uso sejam distribuídos para diversos analistas ou desenvolvedores, se for o caso.

Uma regra a ser observada, é que cada pacote de casos de uso deve conter aproximadamente 3 a 10 unidades menores (casos de uso, atores ou outros pacotes). A tabela abaixo mostra algumas sugestões de como vários pacotes usam determinado número de casos de uso e atores. As quantidades se sobrepõem, porque é impossível determinar definições exatas.

- 0-15: Não há necessidade de usar pacotes de casos de uso.
- 10-50: Usar um nível de pacotes de casos de uso.
- > 25: Usar dois níveis de pacotes de casos de uso.

## 2.2 Especificação dos Casos de Uso de Análise

### 2.2.1 Estruturação

A estruturação da especificação deve seguir o modelo **Modelo de Análise – UC** citado no roteiro desta fase e seu detalhamento deve ser estruturado da seguinte forma:

#### 2.2.1.1 Quadro de informações

O quadro de informações deverá ser preenchido com as seguintes informações:

- **Fase:** Informar o nome da fase, no caso, Análise conforme o modelo.
- **Nome do Caso de Uso:** Informar o nome do caso de uso conforme consta no Diagrama de Casos de Uso e o nome do arquivo de especificação do caso de uso conforme padrão de nomenclatura exigido. Certifique-se que o nome do arquivo da especificação conste na capa, quadro de informações e rodapé.
- **Pacote:** Informar de qual pacote o caso de uso faz parte. Se o sistema possuir somente um pacote, denomine-o como pacote “Único”, sem numeração.
- **Iteração:** Informar de qual iteração esta especificação faz parte. Este item não é

obrigatório somente se o sistema não estiver seguindo o Processo Unificado.

- **Atores:** Informar qual ou quais são os atores que irão interagir com o caso de uso. O ator citado no caso de uso deve estar em conformidade com o que consta no Diagrama de Casos de Uso.
- **Descrição:** Descrever de forma clara e breve qual é o objetivo do caso de uso, ou seja, resumir brevemente a finalidade do caso de uso referenciando atores envolvidos.
- **Informações Adicionais:** Preencher com informações adicionais que sejam relevantes ao objetivo descrito no item acima.
- **Requisitos Atendidos:** Descrever todos os requisitos que este caso de uso estará atendendo. Este item é importante para que o analista, ao longo do projeto, verifique se todos os requisitos comportamentais do sistema foram atendidos.

### 2.2.1.2 Corpo da Especificação

O corpo da especificação deverá ser preenchido com as seguintes informações:

#### 1 Fluxo de Eventos **(OBRIGATÓRIO)**

[O fluxo de eventos será composto por duas partes principais **Fluxo Principal** e **Fluxos Alternativos**.

O fluxo principal deve abordar o que ocorre quando o caso de uso é acionado.

Os fluxos de eventos alternativos devem abordar o comportamento opcional ou excepcional em relação ao comportamento normal.]

#### 1.1 Fluxo Principal **(OBRIGATÓRIO)**

[Deve descrever o diálogo principal entre o ator e o sistema. Deve descrever como o caso de uso começa e termina.

Exemplo de como começar um caso de uso:

1. Este caso de uso se inicia quando o ator Usuário seleciona a opção de menu Cadastro->Manter Fornecedor.

O fluxo principal deve abordar o que ocorre quando o caso de uso é disparado e levar em consideração os passos que o ator deve executar

para realizar o caso de uso.

Durante a definição dos passos do fluxo principal poderão ser identificados fluxos **opcionais**, que **poderão** ser especificados como **Fluxos Alternativos** ou como subfluxos do próprio fluxo principal. Já os fluxos de exceções identificados deverão ser especificados como **Fluxos Alternativos.**]

O Fluxo principal deverá conter as chamadas para os fluxos alternativos, se existirem.

## 1.2 Fluxos Alternativos (OPCIONAL)

[Os Fluxos Alternativos são desvios do fluxo principal, alguns destes desvios voltarão ao fluxo principal e outros finalizarão o caso de uso.

A organização dos fluxos alternativos poderá ser particionada em subfluxos, se necessário para melhorar a compreensão.]

Um fluxo alternativo pode chamar outro fluxo, mas deve-se evitar muitos encadeamentos de fluxos pois isso dificulta a leitura e o entendimento que se quer ter.

## 2 CONSIDERAÇÕES (OPCIONAL)

[Espaço reservado para considerações gerais que não se apliquem aos demais tópicos do documento.]

Certifique-se de que nas considerações não hajam passos que deveriam estar no fluxo de eventos. As considerações são como comentários ou observações, não serão implementadas.

## 3 PRÉ-CONDIÇÕES (OPCIONAL)

[Define o estado em que o sistema deve estar antes do caso de uso ser realizado. Por exemplo:

“O usuário deverá estar logado.”

“O Estabelecimento deve estar devidamente cadastrado.” ]

As pré-condições são informações que o sistema pode assegurar que



serão verdadeiras ao iniciar o caso de uso, elas não fazem parte do fluxo de eventos.

Há duas situações que sinalizam pré-condição. A mais comum é a do usuário logado. A outra é quando um caso de uso é acionado por outro e espera que o caso de uso chamador tenha montado uma condição em particular que ele possa confiar.

#### 4 PÓS-CONDIÇÕES (OPCIONAL)

[Define os possíveis estados em que o sistema irá se encontrar após o término do caso de uso. ]

As pós-condições são asserções que se aplicam ao final da execução do caso de uso, elas mostram o objetivo alcançado pelo ator.

É comum alguns analistas colocarem na pós-condição a frase “gravar log de operação”, ou algo similar. O tratamento de log deve constar como um passo do fluxo de eventos, pois ele será implementado, ou seja ele é uma ação a ser realizada em algum momento. Ao contrário das pós-condições que não são implementadas.

#### 2.2.1.3 Lembretes e Dicas para especificar um Caso de Uso

##### **Lembretes:**

1. Lembre-se que um ator tem um objetivo, e o sistema deve ajudar o ator a atingir este objetivo. É no caso de uso que a interação do ator com o sistema é descrita, como uma seqüência de ações. As ações devem abordar o cenário de sucesso e o tratamento de todas as falhas.

Cenário: é uma seqüência de ações e interações que ocorrem sob certas condições.

2. Procure iniciar a especificação com o cenário de sucesso, ou seja, escrevendo todos os passos que levam o ator a alcançar o objetivo. Depois, aos poucos, vá incluindo as exceções. Desta forma evita-se criar um caso de uso sem o objetivo principal.
3. **Importante:** Identificar todas as possibilidades de falha antes de iniciar a construção, pois identificá-las durante a programação será oneroso para o projeto.

4. Verifique se cada caso de uso especificado consta no Diagrama de Casos de Uso.
5. No passo onde é descrito que o sistema exibe uma tela, inclua, logo em seguida, para cada campo da tela a sua especificação. (Ver tópico “Orientações sobre a especificação de Campos em Formulários” deste documento). Evite deixar a especificação dos campos em um item no final do caso de uso, para facilitar a leitura.
6. Para casos de uso <<include>> ou <<extend>> que recebam parâmetros, verifique se os parâmetros passados no caso de uso chamador e recebidos no caso de uso chamado estão especificados, compatíveis e corretos.
7. Independente se o passo que você está especificando é o Fluxo principal ou o alternativo, você estará descrevendo uma das seguintes ações:
  - Uma interação entre o ator e o sistema (“Ator informa o CPF”);
  - Uma validação (“Sistema valida o CPF”);
  - Uma mudança/atualização interna (“Sistema atualiza os dados do cadastro”).

### **Dicas Sobre Estilo:**

1. Escreva os passos numa sequência lógica, conforme acontece a interação ator/sistema.
2. Use gramática simples. Uma sentença mal formada torna o passo difícil de se entender.
3. Mantenha os passos curtos e objetivos. Escrever muito detalhe em tudo deixa o caso de uso extenso e a leitura cansativa e confusa.
4. Numere os passos, isso clarifica a especificação e facilita a comunicação. Citar onde está uma determinada dúvida fica bem mais fácil se o passo estiver numerado.
5. Prefira a especificação dos campos da tela na forma de texto corrido ao invés de usar tabelas. Alguns itens, como validação e população de combos, são extensos e quando colocados em colunas de tabelas ficam com a leitura prejudicada.
6. Alinhe e numere o tratamento de condições: “Se”, “Senão” etc.
7. A cada passo, cite quem vai realizar a ação. Exemplo:
  - 1 Ator: informa os dados solicitados;
  - 2 Sistema: valida os dados.
8. Quando um passo fizer referência a outro, indique o número do passo ao invés da ação

que ele realiza. Exemplo: ”ao clicar em X, executar o passo 5 do fluxo principal” é melhor que “ao clicar em X, exibir a interface Y e chamar o método Z....”.

9. A especificação de caso de uso de análise deve dizer “o quê” o sistema deve fazer(quais serviços serão disponibilizados pelo sistema), e não “como” irá fazer. O “como” fazer será preocupação da especificação de caso de uso de projeto.
10. Todo fluxo deve ter um último passo para indicar como ele termina.

### 2.2.2 Orientação sobre especificação de campos de Formulário

Em um formulário (tela) é possível utilizar vários tipos de campos como: listbox, combobox, textfield, label, radio button, checkbox e textarea. Para cada tipo de campo **deve-se** informar, na especificação de caso de uso, itens que auxiliarão na implementação do mesmo.

A seguir são apresentados alguns exemplos de utilização destes tipos de campos, com seus respectivos itens.

#### 2.2.2.1 Listbox / Combobox

##### a) Escola

**Tipo do Campo:** Listbox

**Obrigatório:** Sim

**Situação:** habilitado

**Popular com:** Listas de Escolas



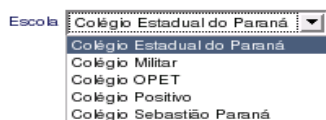
##### a) Escola

**Tipo do Campo:** combobox

**Obrigatório:** Sim

**Situação:** habilitado

**Popular com:** Listas de Escolas

**Observação:**

Escola

- Colégio Estadual do Paraná
- Colégio Militar
- Colégio OPET
- Colégio Positivo
- Colégio Sebastião Paraná

Estabelecer filtros para o item “Popular com”, se necessário.

Ex: **Popular com:** Listas de Escolas filtradas pelo município selecionado.

**2.2.2.2 Textfield****a) CPF**

**Tipo do Campo:** textfield

**Obrigatório:** Sim

**Situação:** habilitado

**Tamanho:** 14

**Máscara:** 999.999.999-99

**Validação:** Utilizar validação para CPF. Caso cpf inválido apresentar a mensagem “CPF inválido”

CPF: 999.999.999-99

**2.2.2.3 Label****a) Nome da Escola**

**Tipo do Campo:** label

**Tamanho:** 50

Nome da escola: Colégio Estadual do Paraná

**2.2.2.4 Radio Button****a) Tipo do Veículo**

**Tipo do Campo:** radio button

**Situação:** habilitado

**Opções:** Carro

Moto

Ônibus  
Caminhão

Tipo do Veículo:

- ☒ Carro
- ☐ Moto
- ☐ Ônibus
- ☐ Caminhão

**Observação:**

**Para opções carregadas em tempo de execução, preencher o item opções da seguinte maneira:**

**Opções:** Popular com a Lista de Tipo de Veículo cadastrada.

#### 2.2.2.5 CheckBox

**a) Usuário Administrador**

**Tipo do Campo:** checkbox

**Situação:** habilitado

☒ Usuário administrador

#### 2.2.2.6 TextArea

**a) Observação**

**Tipo do Campo:** textarea

**Obrigatório:** Não

**Situação:** habilitado

**Tamanho:** 100

**Validação:** alfanumérico

Observações:

#### **2.2.2.7 Hidden**

##### **a) CodReservaPassagem**

Tipo do Campo: hidden

#### **2.2.2.8 Button**

##### **a) Salvar**

Tipo do Campo: button

Situação: habilitado

Visível: sim

#### **2.2.2.9 Link**

##### **a) Liberação de Despesas**

Tipo do Campo: link

Situação: habilitado

### **2.2.3 Orientação para Validação de Campo Obrigatório**

Após a apresentação dos campos da tela deve-se especificar em um único parágrafo a Validação de Campos Obrigatórios conforme exemplo:

Validar o preenchimento dos campos obrigatórios, em caso de não preenchimento emitir a mensagem “O campo {0} deve ser preenchido!”, onde {0} deve ser substituído pelo referido campo.

## 2.2.4 Orientação sobre especificação de mensagens

Todas as mensagens que o sistema deverá emitir ao usuário deverão ser descritas na Especificação de Casos de Uso de Análise, inclusive mensagens emitidas a partir de uma validação de campo de formulário. Por exemplo:

- **Validação de Campo de Formulário:**

- a) **CPF**

- Tipo do Campo: textfield

- Obrigatório: Sim

- Situação: habilitado

- Tamanho: 14

- Máscara: 999.999.999-99

- Validação: Utilizar validação para CPF. Caso cpf inválido apresentar mensagem **“CPF inválido”**

- **Validação de Negócio: Trecho de uma especificação de caso de uso de Análise(Caso de uso Matricular Aluno)**

- “...

- 3 – O usuário informa os dados da matrícula e clica na opção incluir

- 4 – O sistema inclui a nova matrícula se o aluno possuir a idade mínima requerida para se matricular no curso selecionado. Caso o aluno não tenha idade mínima, o sistema não incluirá a matrícula e emitirá a seguinte mensagem **“O Aluno não possui a idade mínima exigida para cursar este curso.**

- ...”

## 2.2.5 Orientação sobre a especificação de casos de uso de relatórios

A especificação de caso de uso de relatórios não difere de uma especificação de caso de uso de formulário. Esta especificação deve abordar o que ocorre quando o caso de uso é acionado, descrevendo mecanismo de seleção do relatório (se houver) e quais campos este relatório apresentará.

O mecanismo de seleção será correspondente à tela de seleção a ser elaborada no artefato protótipo do sistema.

O layout do relatório corresponde aos campos e a distribuição dos mesmos, sendo que:

- Se houver um modelo oficial já utilizado pelo cliente deve-se identificar os campos deste modelo e transcrevê-los para a especificação de caso de uso. Considerar também a cópia do modelo oficial como documentação do sistema, a fim de facilitar a compreensão do layout pelo desenvolvedor.
- Caso não haja um modelo oficial, aconselha-se que este seja prototipado, mesmo que em html, para facilitar a compreensão do layout pelo desenvolvedor, e seus campos transcritos na especificação de caso de uso.

Na especificação de caso de uso também deve ser esclarecido o tipo do relatório (**operacional/online** ou **analítico/batch**). Se for analítico, deve-se informar se o relatório será agendado por data ou se será um agendamento repetitivo.

- **Para agendamento por data**, deve-se informar a data e hora em que o relatório deve ser disparado.
- **Para agendamento repetitivo**, deve-se informar a periodicidade (DIÁRIO, SEMANAL, MENSAL) e o horário a ser disparado.

Independente do tipo de agendamento, deve-se também informar como o relatório será recuperado – através de alguma funcionalidade ou encaminhamento por email. No caso de relatórios enviados por email deve-se configurar também o assunto do email, geralmente colocando o nome do relatório e a data de geração.

Os relatórios analíticos gerados ficam disponíveis no servidor por 15 dias(tempo padrão) mas este prazo pode ser configurado de acordo com a necessidade da aplicação. Sendo assim, se o tempo necessário for maior que o padrão, este deve ser informado na especificação de caso de uso.

É importante ressaltar que de acordo com a arquitetura adotada pela Celepar os relatórios são criados no formato PDF, permitindo que o usuário final possa apenas imprimí-los ou salvá-



los, sem permissão para alterações.

Para maiores detalhes sobre os tipos de relatórios existentes, consultar o documento de apoio ***“Procedimento para implementação de relatórios”***.

## **2.3 Maquete (Identidade visual) / Protótipo do Sistema**

### **2.3.1 Estruturação de formulários de Endereço**

Existe um componente chamado NEOCEP cuja finalidade é operacionalizar o funcionamento de formulários de endereço, utilizando uma base corporativa de endereços. Nas orientações da fase de projeto este componente é explicado com mais detalhes, entretanto, na geração do protótipo do sistema já deve-se levar em consideração se será utilizado este componente, sendo que, seu uso exige que o formulário possua todos os campos que são considerados obrigatórios .

#### **Campos Obrigatórios:**

1. Cep;
2. UF;
3. Município;
4. Bairro;
5. Logradouro;

#### **Campos opcionais:**

6. Número;
7. Complemento;
8. Telefone;
9. Fax;
- ...

Os campos opcionais, como o próprio nome diz, são opcionais e não interferem no funcionamento do componente, sendo assim, outros campos podem ser adicionados sem problema.

Para facilitar o preenchimento dos dados pelo usuário, usando a tecla de navegação TAB, aconselha-se a ordem de disposição no formulário conforme a numeração indicada acima nos campos obrigatórios e opcionais. Esta ordem também é importante devido as formas de pesquisa do componente NEOCEP( 1. informando-se apenas o CEP , 2.informando-se UF + Município + parte do logradouro).

### 2.3.2 Revisão

É importante que o cliente e o analista de sistema aprovem a maquete e o protótipo de telas. As partes interessadas deverão avaliar:

▪ <b>CHECK-LIST</b>	
▪	É cansativo utilizar a interface do software, após várias horas de trabalho?
▪	O símbolo do sistema representa o cliente e o objetivo pelo qual ele está sendo construído?
▪	É definido um padrão de interface? Nesse padrão é definido o estilo de telas de manutenção, botões de comandos, sistema de navegação genérico e segurança, <i>template</i> de navegação, mensagens de erros e help's das funcionalidades?
▪	Atende às expectativas do cliente?

## 2.4 Homologação

Este tópico tem por objetivo auxiliar os analistas a verificarem se todos os artefatos a serem enviados para homologação estão completos, ou seja, se todos os produtos obrigatórios foram gerados e estão de acordo com o processo metodológico.

Analisar os produtos de acordo com o Roteiro e Orientações da fase.

### 2.4.1 Dicas

- Verificar as referências cruzadas: O nome utilizado para o caso de uso consta da mesma forma em todos os lugares onde é citado: no diagrama de UCS, na sua especificação de caso de uso(capa, cabeçalho e rodapé), bem como, as referências a casos de uso por *include* ou *extend* que existirem.
- Glossário: Verificar se a aplicação utiliza siglas, termos ou expressões pertinentes ao negócio que não são de domínio público. Neste caso deve ser criado um glossário para

explicar seus significados.

### **3 REFERÊNCIAS**

LARMAN, CRAIG. Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientados a objetos e ao processo unificado, Porto Alegre: Bookman, 2. edição, 2004

Cockburn, Alistair. Escrevendo Casos de Uso Eficazes, Bookman, 2005.

Rational Unified Process disponível em: <http://www.wthreex.com/rup/>