

# PROBUS

## **Guia de Referência Disciplina de Requisitos – Probus**

Este guia apresenta a definição e descrição de atividades, tarefas, papéis e produtos de trabalho da disciplina.

**Versão: 1.03**

## Guia de Referência Disciplina Requisitos

<b>1. Introdução</b>	<b>3</b>
1.1 Objetivos da Disciplina .....	3
1.2 Fatores Críticos de Sucesso da Disciplina .....	3
1.3 Relacionamentos da Disciplina .....	4
<b>2. Fluxos de Trabalho da Disciplina</b>	<b>5</b>
2.1 Atividade “Definir o Sistema” .....	5
2.1.1 Tarefa “Desenvolver a Visão do Sistema” .....	7
2.1.2 Tarefa “Localizar Atores e Casos de Uso” .....	11
2.1.3 Tarefa “Desenvolver Especificações Suplementares” .....	14
2.1.4 Tarefa “Capturar Vocabulário Comum” .....	16
2.1.5 Tarefa “Gerenciar Dependências dos Requisitos” .....	18
2.2 Atividade “Refinar a Definição do Sistema” .....	20
2.2.1 Tarefa “Especificar Caso de Uso” .....	22
2.2.2 Tarefa “Elaborar Interface e Protótipo de Interface de Caso de Uso” .....	24
2.2.3 Tarefa “Refinar Especificações Suplementares” .....	27
<b>3. Tabela RACI</b>	<b>29</b>
<b>4. Produtos de Trabalho (Artefatos) da Disciplina</b>	<b>30</b>
4.1 Tabela de Entrada/Saída.....	32
<b>5. Onde Posso Aprender Mais sobre a Disciplina?</b>	<b>33</b>
<b>Apêndice I – Tabela Relacionando Tarefas e Objetivos da Disciplina</b>	<b>34</b>

## **1. Introdução**

O PRODUS-TSE é um processo unificado de desenvolvimento de *software* com base no RUP criado para atender às necessidades da STI/TSE em organização e gestão das atividades e produtos de desenvolvimento e manutenção de *softwares*. Existem quatro fases (Admissão, Formulação, Construção e Entrega) e sete disciplinas (Requisitos; Análise e Projeto; Implementação; Teste; Implantação; Configuração e Gerenciamento de Mudanças; e Ambiente).

Uma disciplina é um conjunto de atividades relacionadas a uma “área de interesse” importante em toda a iniciativa de desenvolvimento e manutenção de *software*. As disciplinas representam o eixo vertical do PRODUS, que agrupa as atividades de maneira lógica, por natureza.

### **1.1 Objetivos da Disciplina**

Os objetivos da disciplina de Requisitos são:

- Estabelecer e manter concordância com as partes interessadas sobre o problema organizacional e o que o sistema deve fazer para tratá-lo.
- Definir os limites do sistema.
- Oferecer aos desenvolvedores do sistema uma base para a compreensão dos requisitos do sistema.
- Fornecer uma base para planejamento técnico e gerencial das iterações de desenvolvimento do sistema.
- Fornecer uma base para estimar e medir o tamanho funcional do sistema.
- Fornecer uma base para estimar custo e tempo de desenvolvimento do sistema.
- Definir uma interface de usuário do sistema, focando nas necessidades e metas dos usuários e dos processos organizacionais do TSE.

### **1.2 Fatores Críticos de Sucesso da Disciplina**

Os principais fatores críticos de sucesso da disciplina de Requisitos são os seguintes:

- Compreender a diferença entre o que as partes interessadas querem e o que realmente precisam.
- Compreender as definições de sucesso das partes interessadas.
- Definir e revisar os requisitos do sistema com as partes interessadas.
- Concentrar-se nos requisitos essenciais de sistema, aqueles que representam maiores realizações de benefícios (valores) para a organização/sociedade.
- Diferenciar o que é requisito essencial de sistema do que é acessório (decorativo) de menor valor de negócio.
- Delimitar os requisitos de sistema no escopo para atender às restrições do cronograma de desenvolvimento do sistema.
- Processo organizacional definido para viabilizar a identificação e validação de requisitos de sistemas estáveis ou completos.

## Guia de Referência Disciplina Requisitos

- Os profissionais de TI compreendem os objetivos do processo organizacional e do sistema em desenvolvimento, bem como entendem a linguagem (jargão, conceitos, etc.) dos clientes e usuários.
- Tratar as mudanças e aumento de escopo (*scope creep*) de requisitos de sistemas, além do planejado inicialmente, e seus efeitos.
- Educar seus clientes e usuários sobre os custos e acordos (*trade-offs*) de projeto de desenvolvimento do sistema.

### 1.3 Relacionamentos da Disciplina

A disciplina de requisitos está relacionada com outras disciplinas do PRODUS-TSE:

- A disciplina de **Análise e Projeto** tem como entrada principal os requisitos de sistema.
- A disciplina de **Teste** verifica e valida o sistema construído quanto a seus requisitos.
- A disciplina de **Configuração e Gerenciamento de Mudança** fornece o mecanismo de controle de alterações dos requisitos do sistema.
- A disciplina de **Ambiente** desenvolve e mantém os produtos de trabalho adaptados, bem como ferramentas e ambientes de TI utilizados pelas equipes durante as atividades de engenharia dos requisitos.

## 2. Fluxos de Trabalho da Disciplina

O fluxo de trabalho representado pela figura sugere uma sequência de execução das atividades a serem realizadas para viabilizar o alcance dos objetivos da disciplina.



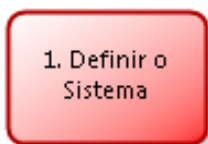
Figura 1 – Fluxo de trabalho da disciplina de Requisitos

### Nota:

Os significados das notações gráficas BPMN utilizadas nos fluxos de trabalho neste guia são apresentados no guia de referência – Notação Gráfica BPMN.

### 2.1 Atividade “Definir o Sistema”

#### 1 Definir o Sistema



#### Objetivos:

- Obter acordo com as partes interessadas sobre a definição do problema organizacional que precisa ser resolvido por um sistema informatizado.
- Buscar entender quais são as reais necessidades das partes interessadas e o que elas desejam a partir de uma solução do sistema.
- Propor uma solução do sistema que atenda às necessidades das partes interessadas
- Refinar a Visão do Sistema para um melhor registro das capacidades do sistema.
- Definir o escopo, em alto nível, dos requisitos funcionais do sistema em termos de casos de uso e atores.
- Descrever, em alto nível, os requisitos específicos que não serão descritos por meio de caso de uso.

#### Descrição da Atividade:

Para realizar essa atividade, é necessário executar as seguintes tarefas:

- 1.1. Desenvolver a visão do sistema.
- 1.2. Localizar atores e casos de uso.
- 1.3. Desenvolver especificações suplementares.
- 1.4. Captar um vocabulário comum (tarefa comum das atividades da disciplina de Requisitos).
- 1.5. Gerenciar dependências de requisitos (tarefa comum das atividades da disciplina de Requisitos).

## 1 Definir o Sistema

### Tarefas da Atividade:

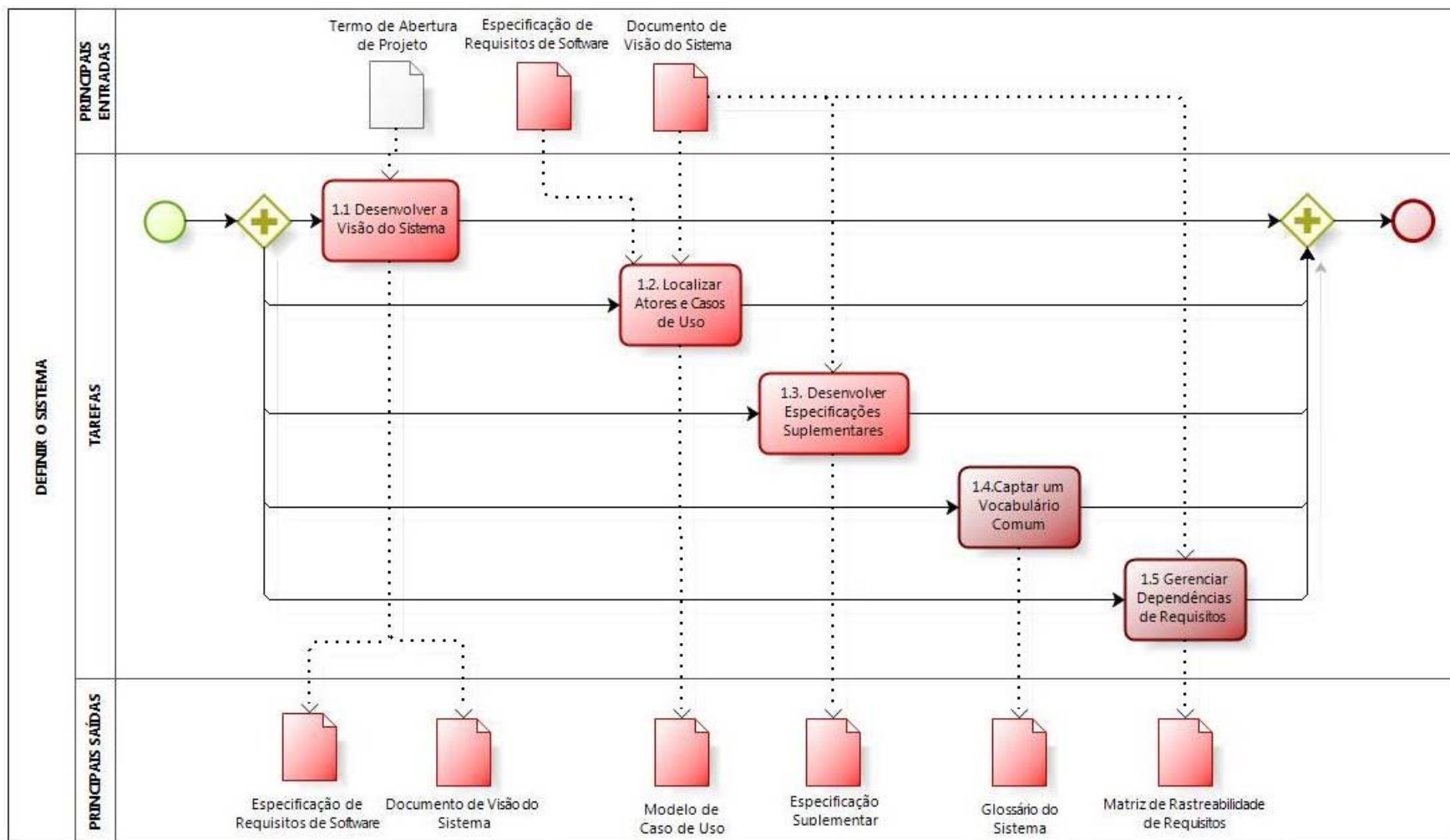


Figura 2 – Fluxo de trabalho da atividade "Definir o Sistema"

## 1 Definir o Sistema

### Orientadores de riscos (*Risk Driver*):

Riscos de não realizar essa atividade:

- Problemas de negócio ou tecnológicos não são resolvidos, de forma eficaz e eficiente, pelo sistema.
- Necessidades das principais partes interessadas e da organização não atendidas pela solução de sistema.
- Retrabalho para adequação do sistema às reais necessidades das principais partes interessadas e da organização.
- Escopo não definido do sistema.
- As principais partes interessadas e a equipe do projeto possuem diferentes visões da funcionalidade desejada do sistema devido à ausência de uma visão geral do comportamento desejado do sistema que possa ser usado como base para o acordo.
- Estimativas não confiáveis do custo e o tempo de desenvolvimento do sistema devido à ausência de uma base confiável para esse fim.
- Planejamentos não confiáveis do conteúdo técnico das iterações de desenvolvimento devido à ausência de uma base confiável para esse fim.
- O sistema não apresentar alguma característica considerada importante pelas principais partes interessadas.
- Arquitetura do *software* inapropriada devido à não identificação de algum requisito não funcional importante do sistema.

### 2.1.1 Tarefa “Desenvolver a Visão do Sistema”

#### 1.1 Desenvolver a Visão do Sistema

1.1 Desenvolver a  
Visão do Sistema

##### Objetivos:

- Estabelecer um acordo com as partes interessadas sobre qual(is) problema(s) precisa(m) ser resolvido(s).
- Identificar as principais partes interessadas do sistema e suas necessidades.
- Estabelecer um acordo com as partes interessadas sobre a priorização das necessidades identificadas
- Definir os limites (escopo) do sistema.
- Definir e descrever as principais características do sistema.

##### Entradas (Produtos Requeridos)

- |  |             |
|--|-------------|
| ▪ Termo de Abertura do Projeto                             | Obrigatória |
| ▪ Termo Simplificado de Abertura e Planejamento do Projeto | Obrigatória |
| ▪ Plano de Ação  | Obrigatória |

##### Critério de Entrada

Critério obrigatório a ser atendido para que a atividade seja iniciada:

- Projeto ou iniciativa aprovada pela Comissão Técnica de Tecnologia da Informação – CTTI.

## 1.1 Desenvolver a Visão do Sistema

### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- **Identificar partes interessadas**  
A identificação das partes interessadas no sistema é um dos primeiros passos para a compreensão completa do problema organizacional e para idealizar um sistema que resolva esse problema. Consiste na tarefa de identificar todas as pessoas ou unidades do TSE que podem ser afetadas pelos resultados do desenvolvimento ou manutenção do sistema.  
A maioria dos sistemas tem um grande número de partes interessadas, logo, é necessária a definição de um dono do produto, ou *Product Owner* (PO), que representa os interesses de todas as partes interessadas, define as funcionalidades do produto e prioriza os itens de *Product Backlog* (lista ordenada de tudo que deve ser necessário no produto).
- **Identificar as necessidades das partes interessadas**  
Colete as informações das partes interessadas sobre o que eles esperam ou desejam que o sistema execute.
- **Adquirir acordo sobre o problema a ser resolvido**  
Evite se apressar na definição das características e requisitos do sistema. Primeiro obtenha um consenso com as partes interessadas sobre o problema a ser resolvido, perguntando o que eles enxergam como problema. Então investigue suas causas raízes (o "problema atrás do problema"). Formule e aprove a descrição do problema.
- **Definir limites do sistema**  
Encontre e defina a linha que divide a solução e o mundo real que a cerca. Identifique as interfaces, bem como as informações de entrada e saída trocadas com os usuários, máquinas ou sistemas.  
Colabore com o gerente de projeto e o arquiteto, visto que as decisões relativas aos limites do sistema terão um grande impacto no custo, no cronograma e na arquitetura do sistema.  
A construção de modelo de caso de uso é uma técnica que pode ser útil na definição dos limites do sistema. Para mais informações, veja a tarefa de requisitos 2.2 – Localizar Atores e Casos de Uso.
- **Identificar restrições no sistema**  
Considere as várias fontes de restrições que podem afetar o *design* ou o próprio projeto. Colabore com o gerente de projeto e o arquiteto de *software*, visto que as decisões relativas às restrições no sistema terão um grande impacto no custo, no cronograma e na arquitetura do sistema.
- **Defina as características do sistema**  
Trabalhe com as principais partes interessadas, com base nos benefícios listados na(s) descrição(ões) do(s) problema(s), para capturar uma lista das características que eles querem no sistema, descrevendo-as resumidamente e registrando os atributos para ajudar a definir suas prioridades e suas situações (*status*) gerais no projeto. Atualize o documento de Visão do Sistema para capturar as características identificadas e seus atributos.
- **Obter consenso da visão do sistema**  
Conduza uma revisão da visão do sistema com as partes interessadas e com a equipe de desenvolvimento para assegurar o consenso, avaliar a qualidade e identificar as mudanças necessárias.



## 1.1 Desenvolver a Visão do Sistema

### Critérios de Aprovação

Critérios obrigatórios a serem atendidos para que a atividade seja considerada finalizada:

- Problema organizacional ou tecnológico a ser resolvido definido e acordado com as partes interessadas.
- Necessidades organizacionais ou tecnológicas a serem atendidas pelo sistema, definidas, priorizadas e acordadas com as partes interessadas.
- Principais características do sistema identificadas, descritas e acordadas com as partes interessadas.
- Limites da solução de sistema proposto estabelecido e acordado com as partes interessadas.
- Restrições no sistema identificadas.

### Saída (Resultados):

▪ Especificação de Requisitos de <i>Software</i>	Opcional	Produto Gerado
▪ Documento de Visão do Sistema	Obrigatória	Produto Gerado

**Padrões e Normas:** Não se aplica.

### Responsável:

- Analista do Sistema

### Principais Participantes:

- Dono do Produto (PO)
- Partes Interessadas
- Gerente de Projeto
- Arquiteto de *Software*
- Administrador de Dados (Sead)

### Ferramentas:

- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

### Erros comuns de execução da tarefa:

- Definir as características e requisitos do sistema antes de se obter um consenso na definição do problema organizacional a ser resolvido pelo sistema.
- Não identificar as partes interessadas que não são usuárias do sistema.
- Não identificar ou compreender as causas-raiz do problema (o problema atrás do problema) que afeta as partes interessadas.
- Não mostrar vínculo entre as características do sistema com as necessidades das partes interessadas.
- Definir os requisitos funcionais antes das características do sistema.
- Esquecer os requisitos não funcionais e as restrições no sistema. Estes, geralmente são os maiores custos de desenvolvimento.
- Não identificar as restrições não técnicas (políticas, econômicas e ambientais) que normalmente conduzem a problemas na entrega e operação do sistema.
- Não identificar e não resolver conflitos de características, requisitos não funcionais e restrições do sistema.

### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

#### Contratos de Infraestrutura Tecnológica, de Sustentação de Software e de Apoio à Gestão de TI

Atividades relacionadas nos catálogos de serviços:

- Sustentação de *Software*
  - Subprocesso 1.1: Levantar Requisitos e Especificar Solução
    - \* 1.1.1 Elaborar documentação de requisitos

### 1.1 Desenvolver a Visão do Sistema

<b>Manutenção de Sistema</b>	<p>De acordo com a norma internacional ISO/IEC 14764:2006 (<i>Software Engineering -- Software Life Cycle Processes -- Maintenance</i>) e a norma brasileira ABNT NBR ISO/IEC 12207:2009 (Engenharia de sistemas e <i>software</i> – Processos de ciclo de vida de <i>software</i>), antes da implementação da modificação do sistema, o mantenedor deve analisar o relatório de problema ou pedido de modificação segundo o seu impacto na organização, no sistema existente e nos sistemas com os quais interage.</p> <p>Deve-se verificar se a solicitação de modificação altera a visão do sistema. Caso positivo, o impacto na visão deve ser comunicado e aprovado pelas partes interessadas. Após a aprovação da alteração da visão, o documento associado deve ser atualizado, aprovado e distribuído/disponibilizado.</p>
<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	Sem comentário adicional para esse cenário.
<b>Fábrica de Software</b>	Sem comentário adicional para esse cenário.

#### **Orientadores de riscos (*Risk Driver*):**

Riscos de não realizar essa tarefa:

- Problemas de negócio ou tecnológicos não são resolvidos, de forma eficaz e eficiente, pelo sistema.
- Necessidades das partes interessadas e da organização não atendidas pelo sistema:
- A síndrome "Sim, mas" na entrega do sistema.
- "Sim, {ele atende aos requisitos}, mas {não resolve o meu problema}."
- Retrabalho para adequação do sistema às reais necessidades das principais partes interessadas e da organização.
- Necessidades das partes interessadas não serem atendidas, de forma eficaz e eficiente, pelas características do sistema. O sistema não apresentar alguma característica considerada importante pelas principais partes interessadas:
- A síndrome "Sim, mas" na entrega do sistema.
- "Sim, {ele atende aos requisitos}, mas {não resolve o meu problema}."
- Problemas organizacionais não serem resolvidos pelas características do sistema.
- Retrabalho para adequação do sistema às reais necessidades das principais partes interessadas e da organização.
- Aumento dos custos de desenvolvimento devido à não identificação de restrições e de requisitos não funcionais no sistema que conduziram a uma arquitetura de *software* inadequada, que necessitará ser ajustada para atender às restrições e aos requisitos não funcionais tardiamente identificados.
- Dificuldade na gestão do escopo do trabalho devido à ausência da definição consensual entre as partes envolvidas do que está dentro e do que está fora das fronteiras do sistema.

#### **Informações adicionais:**

- As restrições reunidas nessa tarefa serão a entrada inicial para as restrições de *design* definidas nas especificações suplementares.
- Características de sistema são utilizadas para estabelecer comunicação e entendimento em um alto nível de abstração, mas não são suficientes para derivar a descrição completa do sistema.

## 2.1.2 Tarefa “Localizar Atores e Casos de Uso”

### 1.2 Localizar Atores e Casos de Uso

#### 1.2. Localizar Atores e Casos de Uso

#### Objetivo:

- Definir o escopo do sistema – o que será tratado pelo sistema e o que será tratado fora dele.
- Definir quem e o que interagirá com o sistema.
- Definir a funcionalidade do sistema na visão do usuário.

#### Entradas (Produtos Requeridos)

- |  |          |
|--|----------|
| ▪ Especificação de Requisitos de <i>Software</i> | Opcional |
| ▪ Documento de Visão do Sistema                  | Opcional |

#### Critério de Entrada

Critério obrigatório a ser atendido para que a atividade seja iniciada:

- Principais características do sistema identificadas.

#### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- Localizar atores  
Identifique grupos de usuários, *softwares* e *hardware* com os quais o sistema deverá interagir. Para cada ator identificado, forneça uma descrição resumida que inclua a área de responsabilidade de cada um e o uso que os atores farão do sistema.
- Localizar casos de uso  
Identifique os casos de uso que consideram o que cada ator exige do sistema. O sistema existe apenas para seus usuários, portanto, deve se basear em suas necessidades.  
Cada caso de uso deve ter um nome que indique o que é alcançado, na visão de negócio, pelas suas interações com os atores, bem como deve ter uma breve descrição.
- Descrever como é a interação entre atores e casos de uso  
Ao localizar um caso de uso, você deve estabelecer quais atores vão interagir com ele e, se necessário, definir a navegabilidade da associação de comunicação, indicando quem inicia a comunicação na interação.
- Empacotar casos de uso e atores  
Se o número de atores ou casos de uso for muito grande, na ordem de dezenas, divida-os em pacotes de casos de uso para simplificar a manutenção, compreensão e a atribuição de responsabilidades do modelo de casos de uso. Neste último caso, permitindo que os desenvolvedores sejam responsáveis por pacotes de casos de uso ou de atores.
- Apresentar o modelo de casos de uso em diagramas  
É possível ilustrar relacionamentos entre casos de uso e atores, bem como entre casos de uso relacionados, em diagramas de casos de uso, que fornecem visões diferentes do modelo de casos de uso. Existem várias formas de se observar o modelo de caso de uso do sistema em construção, geralmente cada parte interessada ressalta funcionalidades que lhe interessam e omite aquelas que não são relevantes.
- Avaliar os modelos de caso de uso  
Nessa etapa, consulte o modelo de casos de uso para verificar se o trabalho está na direção certa, mas não efetue uma revisão detalhada. Para recomendações específicas sobre o que avaliar, consulte a lista de verificação (*checklist*) do modelo de caso de uso.
- Aprovar os modelos de caso de uso  
É importante que as pessoas de fora da equipe de desenvolvimento (por exemplo, usuários e clientes), principalmente as partes interessadas, aprove o modelo de casos de uso nessa etapa.

## 1.2 Localizar Atores e Casos de Uso

### Critérios de Aprovação

Critérios necessários de serem atendidos para que a atividade seja considerada finalizada:

- Modelo de caso de uso aprovado pelas partes interessadas.
- Todos os requisitos funcionais mapeados para ao menos um caso de uso.
- Todos os requisitos não funcionais que devem ser atendidos por casos de uso específicos mapeados para esses casos de uso.

### Saída (Resultados):

- |                          |             |                |
|--------------------------|-------------|----------------|
| ▪ Modelo de Casos de Uso | Obrigatória | Produto Gerado |
|--------------------------|-------------|----------------|

### Padrões e Normas

Não se aplica.

### Responsável:

- Analista do Sistema

### Principais Participantes:

- Dono do Produto (PO)
- Partes Interessadas
- Gerente de Projeto

### Ferramentas:

- Ferramenta CASE de modelagem UML
- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

### Erros comuns de execução da tarefa:

- Decomposição funcional do sistema.
- O modelo de caso de uso contém comportamento supérfluo, não exigido por um ator.
- Alguns relacionamentos entre os casos de uso não são necessários.
- Modelo grande ou com partes com responsabilidades atribuídas para equipes diferentes não utiliza pacotes de casos de uso.
- Requisitos funcionais não mapeados para ao menos um caso de uso.
- Modelo de casos de uso não apresenta o comportamento do sistema de modo claro.

### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

<b>Contratos de Infraestrutura Tecnológica, Sustentação de Software e de Apoio à Gestão de TI</b>	<b>de</b>	Atividades do catálogo de serviços:
	<b>de</b>	▪ Sustentação de <i>Software</i>
	<b>de</b>	▪ Subprocesso 1.1: Levantar requisitos e especificar solução
		* 1.1.11 Elaborar relatório sintético de modelo de caso de uso (MUC)

## 1.2 Localizar Atores e Casos de Uso

<b>Manutenção de Sistema</b>	De acordo com a norma internacional ISO/IEC 14764:2006 ( <i>Software Engineering -- Software Life Cycle Processes -- Maintenance</i> ) e a norma brasileira ABNT NBR ISO/IEC 12207:2009 (Engenharia de sistemas e software – Processos de ciclo de vida de software), antes da implementação da modificação do sistema, o mantenedor deve analisar o relatório de problema ou pedido de modificação segundo o seu impacto na organização, no sistema existente e nos sistemas com os quais interage. Deve-se verificar se a solicitação de modificação altera a relação de casos de uso e atores, seja por meio de inclusão, alteração ou exclusão. Caso positivo, o impacto no escopo do sistema deve ser comunicado e aprovado pelas partes interessadas. Após a aprovação da alteração no escopo, o modelo de caso de uso deve ser atualizado, aprovado e distribuído/disponibilizado.
<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	Sem comentário adicional para esse cenário.
<b>Fábrica de Software</b>	O modelo de caso de uso é importante para subsidiar a determinação da fronteira da aplicação na contagem de pontos de função, métrica utilizada na remuneração dos serviços da contratada.

### **Orientadores de riscos (*Risk Driver*):**

Riscos de não realizar essa tarefa:

- Escopo não definido do sistema.
- As principais partes interessadas e a equipe do projeto possuírem diferentes visões da funcionalidade desejada do sistema devido à ausência de uma visão geral do comportamento desejado do sistema que possa ser usado como base para o acordo.
- Estimativas não confiáveis do custo e o tempo de desenvolvimento do sistema devido à ausência de uma base confiável para este fim.
- Planejamentos não confiáveis do conteúdo técnico das iterações de desenvolvimento devido à ausência de uma base confiável para este fim.

### **Informações adicionais:**

Sem informações adicionais para essa tarefa.

## 2.1.3 Tarefa “Desenvolver Especificações Suplementares”

### 1.3 Desenvolver Especificações Suplementares

1.3. Desenvolver  
Especificações  
Suplementares

#### Objetivo:

- Capturar os requisitos que definem atributos de qualidade do sistema, os quais não se aplicam a casos de uso específicos.

#### Entradas (Produtos Requeridos)

- Documento de Visão do Sistema Opcional

#### Critério de Entrada

Critério obrigatório a ser atendido para que a atividade seja iniciada:

- Principais características do sistema identificadas

#### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- Capturar os requisitos funcionais que não são representados por caso de uso  
Embora não seja comum, identifique os requisitos funcionais que não podem ser aplicados a casos de uso específicos, por exemplo, reportar, auditar, imprimir, autenticar (*login*), etc. Tais requisitos funcionais devem ser documentados na especificação suplementar.
- Capturar qualidades do sistema (requisitos não funcionais)  
Identifique os requisitos não funcionais do sistema que têm influência no grau de satisfação do cliente, dos usuários e das outras partes interessadas do sistema, em que as necessidades do cliente e dos usuários são atendidas ou excedidas em relação às expectativas.
- Capturar restrições  
Identifique as restrições de projeto do sistema.
- Capturar requisitos de conformidade  
Identifique os requisitos de conformidade de padrões (padrões regulatórios, padrões de codificação ou guias de estilo de interface com o usuário, etc.).
- Capturar requisitos de documentação  
Identifique os requisitos, se houver algum, para a documentação, como, por exemplo, requisitos para ajuda *on-line*, para documentação do usuário final (guias de instalação, guias do usuário, material de treinamento, etc.).

#### Crítérios de Aprovação

Crítérios obrigatórios a serem atendidos para que a atividade seja considerada finalizada:

- Atributos de qualidades do sistema identificados e aprovados pelas partes interessadas, bem como os requisitos de conformidade e de documentação.
- Restrições de *design* (projeto) do sistema identificadas.

#### Saída (Resultados):

- Especificação Suplementar Obrigatória Produto Gerado

**Padrões e Normas:** Não se aplica.

#### Responsável:

- Analista do Sistema

#### Principais Participantes:

- Dono do Produto (PO)
- Partes Interessadas
- Arquiteto de *Software*

### 1.3 Desenvolver Especificações Suplementares

**Ferramentas:**

- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

**Erros comuns de execução da tarefa:**

- Definição de uma regra de negócio.

#### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

<b>Contratos de Infraestrutura Tecnológica, de Sustentação de Software e de Apoio à Gestão de TI</b>	<p>Atividades do catálogo de serviços:</p> <ul style="list-style-type: none"> <li>▪ Sustentação de <i>Software</i> <ul style="list-style-type: none"> <li>▪ Subprocesso 1.1: Levantar requisitos e especificar solução                             <ul style="list-style-type: none"> <li>* 1.1.12 Elaborar especificação suplementar</li> </ul> </li> </ul> </li> </ul>
<b>Manutenção de Sistema</b>	<p>De acordo com a norma internacional ISO/IEC 14764:2006 (<i>Software Engineering -- Software Life Cycle Processes -- Maintenance</i>) e a norma brasileira ABNT NBR ISO/IEC 12207:2009 (Engenharia de sistemas e <i>software</i> – Processos de ciclo de vida de <i>software</i>), antes da implementação da modificação do sistema, o mantenedor deve analisar o relatório de problema ou pedido de modificação segundo o seu impacto na organização, no sistema existente e nos sistemas com os quais interage.</p> <p>Deve-se verificar se a solicitação de modificação altera a relação de requisitos não funcionais, seja por meio de inclusão, alteração ou exclusão. Caso positivo, esta alteração deve ser comunicada e aprovada pelas partes interessadas. Após a aprovação da alteração no escopo, a especificação suplementar deve ser atualizada, aprovada e distribuída/disponibilizada.</p>
<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	Sem comentário adicional para esse cenário.
<b>Fábrica de Software</b>	Sem comentário adicional para esse cenário.

**Orientadores de riscos (*Risk Driver*):**

Riscos de não realizar essa tarefa:

- Necessidades das partes interessadas não serem atendidas, de forma eficaz e eficiente, pelo sistema, devido a não identificação de um atributo crítico de qualidade do sistema esperado pelas partes interessadas.
- Arquitetura do sistema inadequada.
- Retrabalho para ajustar a arquitetura do sistema para atendimento de requisitos não funcionais críticos do sistema.
- Infraestrutura de produção inapropriada para suportar a operação do sistema devido à não identificação de um atributo crítico de qualidade do sistema que exige capacidades e configurações específicas dos recursos computacionais da infraestrutura de TI do TSE.

**Informações adicionais:**

- Esses tipos de requisitos são muito importantes para determinar a complexidade do sistema.
- Essas restrições são muito importantes para determinar a arquitetura adequada do sistema para atender às necessidades das partes interessadas.



## 2.1.4 Tarefa “Capturar Vocabulário Comum”

### 1.4 Capturar Vocabulário Comum

1.4.Captar um  
Vocabulário  
Comum

#### Objetivo:

- Definir um vocabulário comum que possa ser utilizado em todas as descrições textuais do sistema, especialmente os requisitos de sistema.

#### Entradas (Produtos Requeridos)

- |  |          |
|--|----------|
| ▪ Termo Simplificado de Abertura e Planejamento de Projeto | Opcional |
| ▪ Documento de Visão do Sistema                            | Opcional |
| ▪ Modelo de Caso de Uso                                    | Opcional |
| ▪ Especificação Suplementar                                | Opcional |
| ▪ Especificação de Caso de Uso                             | Opcional |
| ▪ Especificação de Regras de Negócio                       | Opcional |

#### Critérios de Entrada

Critérios obrigatórios a serem atendidos para que a atividade seja iniciada: nenhum.

#### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- Localizar termos comuns  
Para encontrar os termos comuns do domínio de negócio, considere os termos utilizados nos requisitos e o conhecimento geral da equipe de desenvolvimento sobre o sistema a ser criado. Concentre-se nos termos que descrevem os seguintes conceitos:
  - Os objetos de negócio que representam conceitos utilizados no trabalho diário do Tribunal ou no ambiente operacional esperado do sistema. Em muitos casos, uma lista de conceitos desse tipo já existe.
  - Os objetos do mundo real que o sistema precisa identificar. Esses objetos ocorrem naturalmente e podem ser, por exemplo: eleitor, candidato, urna, contrato, cargo efetivo, servidor do TSE, beneficiários ou benefícios.
- Avaliar os termos comuns identificados  
Nessa etapa, analise o glossário ou modelo de domínio, com outro(s) membro(s) da equipe, para verificar se o trabalho está indo na direção correta. Não há necessidade de revisá-lo detalhadamente.

#### Critérios de Aprovação

Critérios obrigatórios a serem atendidos para que a atividade seja considerada finalizada:

- Cada termo tem o mesmo significado em todos os produtos de trabalho de requisitos.
- Cada termo do glossário está incluso em alguma parte dos produtos de trabalho de requisitos.
- Os termos são utilizados consistentemente nos produtos de trabalho de requisitos.
- Os relacionamentos e as cardinalidades desses relacionamentos das entidades de negócio do modelo de domínio estão representados coerentemente com a visão de processo organizacional.

#### Saída (Resultados):

- |                               |             |                |
|-------------------------------|-------------|----------------|
| ▪ Glossário do Sistema        | Obrigatório | Produto Gerado |
| ▪ Modelo de Análise (domínio) | Opcional    | Produto Gerado |



## 1.4 Capturar Vocabulário Comum

### Padrões e Normas:

Não se aplica.

### Responsável:

- Analista do Sistema

### Principais Participantes:

- Dono do Produto (PO)
- Partes Interessadas
- Administrador de Dados (Sead)

### Ferramentas:

- Ferramenta CASE de modelagem UML
- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

### Erros comuns de execução da tarefa:

- Definição de regras de negócio.
- Definição de todos os termos de negócios possíveis.
- Relacionamentos incorretos, bem como cardinalidades incorretas, na visão de negócio, entre as entidades de negócios no modelo de domínio.

### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

<b>Contratos de Infraestrutura Tecnológica, de Sustentação de Software e de Apoio à Gestão de TI</b>	Atividades do catálogo de serviços: <ul style="list-style-type: none"> <li>▪ Sustentação de <i>Software</i> <ul style="list-style-type: none"> <li>* 1.1.1 Elaborar documentação de requisitos</li> <li>* 1.1.4 Elaborar modelo de análise (domínio)</li> </ul> </li> </ul>
<b>Manutenção de Sistema</b>	Sem comentário adicional para esse cenário.
<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	Descreva apenas os termos de negócio cruciais para os desenvolvedores e os termos técnicos cruciais para os clientes.
<b>Fábrica de Software</b>	O modelo de domínio completo e atualizado facilita as atividades de estimativas e medições dos tamanhos funcionais dos sistemas em ponto de função.

### Orientadores de riscos (*Risk Driver*):

Riscos de não realizar essa tarefa:

- Problemas ou ineficiência de comunicação dentro da equipe de projeto ou desta com outras unidades do TSE, devido ao uso de termos ambíguos, não claros, incorretos ou incompletos.
- Atrasos imprevistos nas especificações e validações dos requisitos com o dono do produto (PO) e com as principais partes interessadas.
- Retrabalho de projeto técnico (principalmente relacionado aos modelos de dados) e implementação (codificação) do sistema devido às interpretações erradas de termos importantes usados no sistema e no domínio/processo de negócio. Alguns desses problemas de comunicação tornam-se óbvios somente nos estágios finais de teste e validação do sistema e pode ser extremamente caro corrigi-los neste momento.

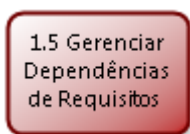
## 1.4 Capturar Vocabulário Comum

### Informações adicionais:

- As definições para os termos do glossário do sistema vêm de diversas fontes, além daquelas identificadas como entrada desta tarefa, tais como documentos de requisitos, especificações e discussões com os principais envolvidos e com os especialistas do domínio de negócio.
- Embora listado como uma saída ou entrada das tarefas associadas com a disciplina de requisitos, o glossário do sistema pode ser atualizado a qualquer momento, por qualquer papel, assim que novos termos sejam identificados pela equipe de projeto.

## 2.1.5 Tarefa “Gerenciar Dependências dos Requisitos”

### 1.5 Gerenciar Dependências dos Requisitos



#### Objetivos:

- Compreender a origem dos requisitos do sistema.
- Avaliar o impacto da mudança em um requisito do sistema.
- Avaliar o impacto de uma falha de um teste nos requisitos do sistema.
- Gerenciar o escopo do projeto.
- Verificar se o aplicativo faz apenas o que era esperado que ele fizesse.
- Verificar se todos os requisitos do sistema foram implementados.
- Gerenciar mudanças nos requisitos.

#### Entradas (Produtos Requeridos)

- |  |          |
|--|----------|
| ▪ Termo de Abertura de Projeto                             | Opcional |
| ▪ Termo Simplificado de Abertura e Planejamento de Projeto | Opcional |
| ▪ Plano de Ação  | Opcional |
| ▪ Mapeamento de Interdependências                          | Opcional |
| ▪ Documento de Visão do Sistema                            | Opcional |
| ▪ Modelo de Caso de Uso                                    | Opcional |
| ▪ Especificação Suplementar                                | Opcional |

#### Critérios de Entrada

Critérios obrigatórios a serem atendidos para que a atividade seja iniciada:

- Nenhum

#### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- Estabelecer e verificar a rastreabilidade  
Estabeleça e mantenha atualizada a rastreabilidade dos requisitos do sistema desde as suas origens para outros artefatos do sistema. Deve-se também estabelecer a rastreabilidade entre os tipos de requisitos.  
Recomenda-se a análise periódica de relatórios de rastreabilidade para garantir que a rastreabilidade seja mantida.
- Gerenciar requisitos variáveis  
Gerencie mudanças de requisitos do sistema, identificando e reportando às partes interessadas os impactos das mudanças.

## 1.5 Gerenciar Dependências dos Requisitos

### Critérios de Aprovação

Critérios obrigatórios a serem atendidos para que a atividade seja considerada finalizada:

- As dependências entre os tipos de requisitos (necessidades do usuário, características do sistema, requisitos de *software*, etc.) foram identificadas.
- Os requisitos do sistema instáveis com alto risco, esforço ou benefício foram identificados.
- As dependências dos tipos de requisitos do sistema para outros artefatos do sistema, principalmente de testes, foram identificadas.
- Impactos de mudanças de requisitos do sistema foram identificados e avaliados.

### Saída (Resultados):

- |  |             |                           |
|--|-------------|---------------------------|
| ▪ Matriz de Rastreabilidade dos Requisitos | Obrigatório | Produto Gerado/Atualizado |
| ▪ Matriz de Dependência entre Elementos    | Opcional    | Produto Gerado/Atualizado |

**Padrões e Normas:** Não se aplica.

### Responsável:

- Analista do Sistema

### Principais Participantes:

-

### Ferramentas:

- Ferramenta CASE de modelagem UML
- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

### Erros comuns de execução da tarefa:

- Não estabelecer a rastreabilidade dos requisitos do sistema para as necessidades da parte interessada.
- Não estabelecer a rastreabilidade dos requisitos do sistema para artefatos de testes.
- Não informar/atualizar os atributos essenciais dos requisitos: “Origem”, “Benefício”, “Esforço”, “Risco”, “Arquitetura – Impacto” e “Estabilidade”.

### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

<b>Contratos de Infraestrutura Tecnológica, Sustentação de Software e de Apoio à Gestão de TI</b>	Atividades do catálogo de serviços: <ul style="list-style-type: none"> <li>▪ Sustentação de <i>Software</i> <ul style="list-style-type: none"> <li>▪ Subprocesso 1.1: Levantar requisitos e especificar solução               <ul style="list-style-type: none"> <li>* 1.1.10 Elaborar matriz de rastreabilidade</li> </ul> </li> </ul> </li> </ul>
<b>Manutenção de Sistema</b>	<p>De acordo com a norma internacional ISO/IEC 14764:2006 (<i>Software Engineering -- Software Life Cycle Processes -- Maintenance</i>) e a norma brasileira ABNT NBR ISO/IEC 12207:2009 (Engenharia de sistemas e <i>software</i> – Processos de ciclo de vida de <i>software</i>), antes da implementação da modificação do sistema, o mantenedor deve analisar o relatório de problema ou pedido de modificação segundo o seu impacto na organização, no sistema existente e nos sistemas com os quais interage. Para determinar se esse impacto é essencial, é recomendado o uso de uma rastreabilidade dos requisitos do sistema.</p> <p>A norma ISO/IEC 14764:2006 informa que, na atividade “implementação da modificação”, o mantenedor deve conduzir a análise e determinar quais documentação, unidades de <i>software</i> e versões destas necessitam ser modificadas. A manutenção da rastreabilidade atualizada dos requisitos do sistema facilita o esforço dessa análise.</p>

### 1.5 Gerenciar Dependências dos Requisitos

<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	Sem comentário adicional para esse cenário.
<b>Fábrica de Software</b>	Para o planejamento e gestão eficaz das demandas dos serviços de desenvolvimento por uma Fábrica de <i>Software</i> , é essencial o controle das dependências dos requisitos, por meio da matriz de rastreabilidade de requisitos, que permite também a avaliação dos impactos das mudanças em requisitos de sistemas já implementados e entregues pela contratada.

#### **Orientadores de riscos (*Risk Driver*):**

Riscos de não realizar essa tarefa:

- Impactos negativos (que podem resultar em dano à imagem ou à credibilidade do TSE ou da STI/TSE) não previstos na mudança em um requisito.
- Dificuldade em identificar se todas as necessidades das partes interessadas foram atendidas ou se os requisitos do sistema foram implementados no sistema.
- Dificuldade em determinar o impacto da falha de um componente crítico do sistema no uso dos requisitos do sistema.

#### **Informações adicionais:**

Sem informações adicionais para essa tarefa.

## 2.2 Atividade “Refinar a Definição do Sistema”

### 2 Refinar a Definição do Sistema

2. Refinar a Definição do Sistema

#### **Objetivo:**

- Detalhar os requisitos a serem desenvolvidos no ciclo atual de desenvolvimento do sistema.

#### **Descrição da Atividade:**

Para realizar essa atividade, é necessário executar as seguintes tarefas:

- 2.1. Especificar caso de uso.
- 2.2. Elaborar interface e protótipo de interface de caso de uso.
- 2.3. Refinar especificações suplementares.
- 1.4. Capturar o vocabulário comum (tarefa comum das atividades da disciplina de Requisitos).
- 1.5. Gerenciar dependências dos requisitos (tarefa comum das atividades da disciplina de Requisitos).

## 2 Refinar a Definição do Sistema

### Tarefas da Atividade:

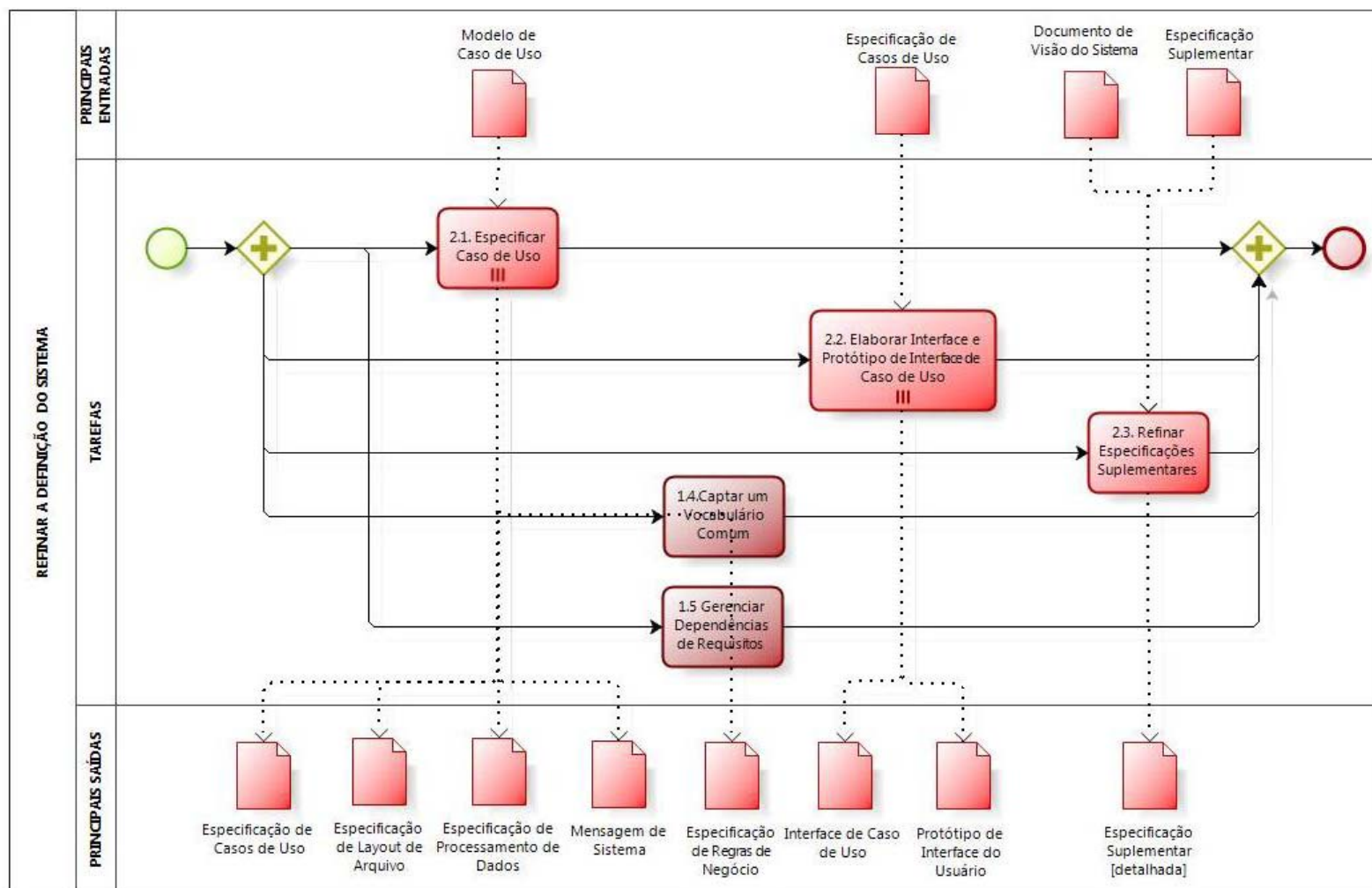


Figura 3 – Fluxo de trabalho da atividade “Refinar a Definição do Sistema”

## 2 Refinar a Definição do Sistema

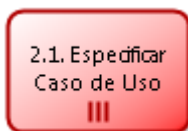
### Orientadores de riscos (*Risk Driver*):

Riscos de não realizar essa atividade:

- Compreensão dos requisitos do sistema não validada pelas partes interessadas para início confiável das atividades técnicas de projeto e codificação.
  - Equipe de desenvolvimento presume os requisitos de sistemas, posteriormente, na entrega do sistema, essas pressuposições podem ser invalidadas.
- Retrabalho e alto custo de correção de defeito (Regra 10 de Myers)

### 2.2.1 Tarefa “Especificar Caso de Uso”

#### 2.1 Especificar Caso de Uso



#### Objetivo:

- Descrever um ou mais dos fluxos do caso de uso em detalhes suficientes para permitir que seja iniciado, com segurança, o desenvolvimento de suas funcionalidades.
- Detalhar o caso de uso para a validação das partes interessadas e a compreensão da equipe de desenvolvimento.

#### Entradas (Produtos Requeridos)

- |                                 |             |
|---------------------------------|-------------|
| ▪ Modelo de Caso de Uso         | Obrigatória |
| ▪ Documento de Visão do Sistema | Opcional    |

#### Critério de Entrada

Critério obrigatório a ser atendido para que a atividade seja iniciada:

- Modelo de caso de uso atualizado, revisado e validado.

#### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- Detalhar o fluxo de eventos  
Descreva o fluxo de eventos da interação do caso de uso com os atores. O fluxo de eventos consiste em um fluxo básico e um ou vários fluxos alternativos.  
Descreva também fluxos incomuns ou excepcionais de uso do sistema. Um fluxo excepcional é um subfluxo do caso de uso que não é compatível com o comportamento normal ou esperado do caso de uso.
- Descrever os requisitos especiais  
Descreva qualquer requisito que possa estar relacionado ao caso de uso, mas que não seja considerado no fluxo de eventos. Provavelmente, esses requisitos serão considerados como do tipo não funcional. Descreva esse requisito na seção “Requisitos Especiais” do documento de caso de uso.
- Descrever regras de negócio  
Regras de negócio é uma declaração de política ou uma condição do processo organizacional que representa restrições estruturais e comportamentais a serem satisfeitas pelo sistema.  
Descreva as regras de negócio que são específicas do caso de uso do próprio documento de especificação. Se as regras forem gerais e abrangerem diversos casos de uso, descreva essas regras em um documento de Especificação de Regras de Negócio e as referencie na especificação do caso de uso em questão.



## 2.1 Especificar Caso de Uso

- Descrever condições prévias  
Descreva o estado do sistema para que o caso de uso possa começar.
- Descrever condições posteriores  
Descreva a lista dos possíveis estados do sistema no fim do caso de uso. Também é importante informar ações que o sistema executa no término do caso de uso, independentemente do que tenha ocorrido no caso de uso.
- Descrever os pontos de extensão  
Se o caso de uso tiver que ser estendido por outro caso de uso, será necessário descrever quais são os pontos de extensão.
- Aprovar a especificação  
Analise e discuta as especificações com as partes interessadas para que tenham uma boa compreensão, concordem com sua descrição e aprovem a especificação de caso de uso e, se existir, também a especificação de regras de negócio.

### Critérios de Aprovação

Critérios obrigatórios a serem atendidos para que a atividade seja considerada finalizada:

- Todas as especificações priorizadas para implementação na iteração atual ou na subsequente ou, ainda, aquelas definidas como arquiteturalmente significantes devem estar descritas e validadas quanto ao padrão, completude e consistência.
- As especificações devem representar a necessidade de negócio declarada para o sistema.

### Saída (Resultados):

▪ Especificação de Casos de Uso	Obrigatória	Produto Gerado
▪ Especificação de <i>Layout</i> de Arquivo	Opcional	Produto Gerado
▪ Especificação de Processamento de Dados	Opcional	Produto Gerado
▪ Especificação de Regras de Negócio	Opcional	Produto Gerado
▪ Mensagem de Sistema	Opcional	Produto Gerado
▪ Especificação de Perfis de Acesso	Opcional	Produto Gerado

**Padrões e Normas:** Não se aplica.

### Responsável:

- Analista do Sistema

### Principais Participantes:

- Dono do Produto (PO)
- Partes Interessadas
- Arquiteto de *Software*
- Administrador de Dados (Sead)

### Ferramentas:

- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

### Erros comuns de execução da tarefa:

- Descrever como problemas específicos e internos do sistema devem ser solucionados ao invés do que é feito no caso de uso.
- Usar jargões próprios da área de TI que não são compreendidos pelas partes interessadas.
- Excesso de detalhes ao descrever a interação do caso de uso com os atores.
- Uso de advérbios, como muito, mais, bastante e outros semelhantes, nas descrições dos fluxos de eventos.
- Frases longas e compostas nas descrições dos fluxos de eventos.
- Utilizar as pós-condições para criar uma sequência de casos de uso.

## 2.1 Especificar Caso de Uso

### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

<b>Contratos de Infraestrutura Tecnológica, de Sustentação de Software e de Apoio à Gestão de TI</b>	<p>Atividades do catálogo de serviços:</p> <ul style="list-style-type: none"> <li>Sustentação de <i>Software</i> <ul style="list-style-type: none"> <li>Subprocesso 1.1: Levantar requisitos e especificar solução                             <ul style="list-style-type: none"> <li>* 1.1.2 Especificar caso de uso</li> </ul> </li> </ul> </li> </ul>
<b>Manutenção de Sistema</b>	<p>De acordo com a norma internacional ISO/IEC 14764:2006 (<i>Software Engineering -- Software Life Cycle Processes -- Maintenance</i>) e a norma brasileira ABNT NBR ISO/IEC 12207:2009 (Engenharia de sistemas e <i>software</i> – Processos de ciclo de vida de <i>software</i>), antes da implementação da modificação do sistema, o mantenedor deve analisar o relatório de problema ou pedido de modificação segundo o seu impacto na organização, no sistema existente e nos sistemas com os quais interage.</p> <p>A atividade “Implementação da modificação”, apresentada na norma 14764, informa que o mantenedor deve utilizar o processo de desenvolvimento, no caso o PRODUS, para implementar as modificações, o que inclui a atualização das especificações dos casos de uso.</p>
<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	<p>Descreva um caso de uso simplificado, abstrato e genérico que capture as intenções do usuário. Descreva os fluxos que foram definidos como arquiteturalmente significantes para o sistema.</p>
<b>Fábrica de Software</b>	<p>Sem comentário adicional para esse cenário</p>

### Orientadores de riscos (*Risk Driver*):

Riscos de não realizar essa tarefa:

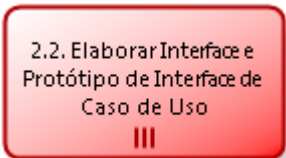
- Compreensão dos requisitos do sistema não validada pelas partes interessadas para início confiável das atividades técnicas de projeto e codificação.
- Retrabalho e alto custo de correção de defeito.
- Premissas assumidas pela equipe de desenvolvimento durante a construção do sistema invalidadas devido à ausência de requisitos detalhados e validados pelas partes interessadas.

### Informações adicionais:

Sem informações adicionais para essa tarefa.

## 2.2.2 Tarefa “Elaborar Interface e Protótipo de Interface de Caso de Uso”

### 2.2 Elaborar Interface e Protótipo de Interface de Caso de Uso

	<p><b>Objetivo:</b></p> <ul style="list-style-type: none"> <li>Identificar e especificar os requisitos dos elementos de interface do caso de uso.</li> <li>Esclarecer os requisitos do caso de uso.</li> <li>Validar premissas sobre o requisito, projeto, funcionalidade, tecnologia, etc.</li> </ul>
---	--

### Entradas (Produtos Requeridos)

- Especificação de Caso de Uso Opcional



## 2.2 Elaborar Interface e Protótipo de Interface de Caso de Uso

### Critérios de Entrada

Critérios obrigatórios a serem atendidos para que a atividade seja iniciada: nenhum.

### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- Identificar as principais janelas da interface com o usuário  
Ao verificar os requisitos considerados em um caso de uso, identifique as principais interfaces do usuário. Essas interfaces referem-se àquelas janelas com as quais o usuário vai interagir. As janelas contêm *menus* e podem conter subjanelas ou formulários. São as janelas pelas quais o usuário navega.
- Construir o protótipo da interface do usuário  
Caso seja necessário, construa um protótipo da interface com o usuário que permita visualizar os aspectos mais importantes da interface do usuário e que ofereça algumas de suas ações/comportamentos significativos do usuário.
- Obter comentários sobre o protótipo da interface do usuário  
É importante trabalhar com os usuários ou usuários potenciais do sistema ao criar o protótipo da interface do usuário. Esse protótipo pode ser utilizado para tratar a usabilidade do sistema, para ajudar a identificar qualquer requisito não descoberto previamente e refinar ainda mais a definição dos requisitos. Os comentários das partes interessadas sobre o protótipo da interface do usuário podem ser obtidos por meio das revisões e testes em foco.
- Descrever a interface de referência para o usuário  
Descreva os elementos da interface do caso de uso com o usuário, principalmente os campos de formulários. Caso seja necessário, em interfaces com campos dinâmicos (valores apresentados em um campo dependem da seleção de determinado valor em outro campo), descreva as regras de apresentação das informações ou de navegação na interface.
- Aprovar especificação da interface de caso de uso  
Analisar e discutir a especificação do caso de uso com as partes interessadas para que elas tenham uma boa compreensão do caso de uso, concordem com sua descrição e aproveem a especificação.

### Critérios de Aprovação

Critérios obrigatórios a serem atendidos para que a atividade seja considerada finalizada:

- Os requisitos dos elementos de interface de usuário identificados, definidos e validados.
- Validadas as premissas sobre o requisito, usabilidade, projeto, funcionalidade, tecnologia, etc. de um caso de uso.

### Saída (Resultados):

- |                                     |             |                |
|-------------------------------------|-------------|----------------|
| ▪ Interface de Caso de Uso          | Obrigatória | Produto Gerado |
| ▪ Protótipo de Interface do Usuário | Opcional    | Produto Gerado |

**Padrões e Normas:** Não se aplica.

### Responsável:

- Analista do Sistema

### Principais Participantes:

- Dono do Produto (PO)
- Partes Interessadas
- Arquiteto de *Software*

### Ferramentas:

- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

## 2.2 Elaborar Interface e Protótipo de Interface de Caso de Uso

### Erros comuns de execução da tarefa:

- Definição de uma regra de negócio.

### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

<b>Contratos de Infraestrutura Tecnológica, de Sustentação de Software e de Apoio à Gestão de TI</b>	<p>Atividades do catálogo de serviços:</p> <ul style="list-style-type: none"> <li>▪ Sustentação de <i>Software</i> <ul style="list-style-type: none"> <li>▪ Subprocesso 1.1: Levantar requisitos e especificar solução <ul style="list-style-type: none"> <li>* 1.1.2 Especificar caso de uso</li> </ul> </li> </ul> </li> </ul>
<b>Manutenção de Sistema</b>	<p>De acordo com a norma internacional ISO/IEC 14764:2006 (<i>Software Engineering -- Software Life Cycle Processes -- Maintenance</i>) e a norma brasileira ABNT NBR ISO/IEC 12207:2009 (Engenharia de sistemas e <i>software</i> – Processos de ciclo de vida de <i>software</i>), antes da implementação da modificação do sistema, o mantenedor deve analisar o relatório de problema ou pedido de modificação segundo o seu impacto na organização, no sistema existente e nos sistemas com os quais interage.</p> <p>A atividade “Implementação da modificação”, apresentada na norma 14764, informa que o mantenedor deve utilizar o processo de desenvolvimento, no caso o PRODUS, para implementar as modificações, o que inclui a atualização das especificações de interfaces do usuário relacionados aos casos de uso.</p>
<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	<p>Uso de um protótipo básico de interface com o usuário, que é um modelo ou protótipo de baixa fidelidade da interface com o usuário do sistema. Representa ideias gerais por trás da interface, mas não os detalhes exatos. Foque nos protótipos de interface de usuário dos casos de uso que foram definidos como arquiteturalmente significantes para o sistema.</p>
<b>Fábrica de Software</b>	<p>As interfaces do usuário são consideradas importantes entradas (insumos) para processo de contagem de pontos de função de um sistema, pois permite a validação das especificações de requisitos do sistema, bem como permite identificar funções do sistema que não estão detalhadas claramente em suas especificações.</p>

### Orientadores de riscos (*Risk Driver*):

Riscos de não realizar essa tarefa:

- Desperdiçar tempo e recursos no desenvolvimento devido a requisitos não descobertos, ambíguos ou incompletos do sistema.
- Premissas inválidas sobre o requisito, projeto, funcionalidade, tecnologia ou ambas.

### Informações adicionais:

Sem informações adicionais para essa tarefa.

## 2.2.3 Tarefa “Refinar Especificações Suplementares”

### 2.3 Refinar Especificações Suplementares

#### 2.3. Refinar Especificações Suplementares

#### Objetivo:

- Validar e detalhar os requisitos que definem atributos de qualidade do sistema e que não se aplicam a casos de uso específicos.

#### Entradas (Produtos Requeridos)

- |                                 |             |
|---------------------------------|-------------|
| ▪ Documento de Visão do Sistema | Opcional    |
| ▪ Especificação Suplementar     | Obrigatória |

#### Crítérios de Entrada

Crítérios obrigatórios a serem atendidos para que a atividade seja iniciada:

- Atributos de qualidades do sistema identificados e aprovados pelas partes interessadas, bem como os requisitos de conformidade e de documentação.
- Restrições de *design* (projeto) ao sistema identificadas.

#### Descrição da Tarefa:

Para realizar essa tarefa, é necessário executar as seguintes etapas:

- Detalhar os requisitos suplementares  
Identifique, refine e valide os requisitos, que não podem ser aplicados a casos de uso específicos, que terão impacto em um trabalho próximo ou que foram definidos como arquiteturalmente significantes, pois alguns requisitos podem necessitar ser esclarecidos ou descritos com mais detalhes, novos requisitos podem ser descobertos enquanto detalhamos os casos de uso e os cenários.

#### Crítério de Aprovação

Crítério obrigatório a ser atendido para que a atividade seja considerada finalizada:

- Os requisitos que não podem ser aplicados a casos de uso específicos identificados, refinados e validados pelas partes interessadas.

#### Saída (Resultados):

- |                             |             |                    |
|-----------------------------|-------------|--------------------|
| ▪ Especificação Suplementar | Obrigatória | Produto atualizado |
|-----------------------------|-------------|--------------------|

#### Padrões e Normas:

Não se aplica.

#### Responsável:

- Analista do Sistema

#### Principais Participantes:

- Dono do Produto (PO)
- Partes Interessadas
- Arquiteto de *Software*

#### Ferramentas:

- Microsoft Office Word 2007 ou versão posterior
- LibreOffice BrOffice Writer 3.3.2 ou versão posterior

#### Erro comum de execução da tarefa:

- Definição de uma regra de negócio.

## 2.3 Refinar Especificações Suplementares

### Comentários adicionais para execução da tarefa em diferentes tipos de cenários

<b>Contratos de Infraestrutura Tecnológica, de Sustentação de Software e de Apoio à Gestão de TI</b>	<p>Atividades do catálogo de serviços:</p> <ul style="list-style-type: none"> <li>Sustentação de <i>Software</i> <ul style="list-style-type: none"> <li>Subprocesso 1.1: Levantar requisitos e especificar solução <ul style="list-style-type: none"> <li>* 1.1.12 Elaborar especificação suplementar</li> </ul> </li> </ul> </li> </ul>
<b>Manutenção de Sistema</b>	<p>De acordo com a norma internacional ISO/IEC 14764:2006 (<i>Software Engineering -- Software Life Cycle Processes -- Maintenance</i>) e a norma brasileira ABNT NBR ISO/IEC 12207:2009 (Engenharia de sistemas e <i>software</i> – Processos de ciclo de vida de <i>software</i>), antes da implementação da modificação do sistema, o mantenedor deve analisar o relatório de problema ou pedido de modificação segundo o seu impacto na organização, no sistema existente e nos sistemas com os quais interage.</p> <p>A atividade “Implementação da modificação”, apresentada na norma 14764, informa que o mantenedor deve utilizar o processo de desenvolvimento, no caso o PRODUS, para implementar as modificações, o que inclui a atualização das especificações suplementares.</p>
<b>Projetos pequenos de baixos graus de complexidades gerenciais e técnicas</b>	Sem comentário adicional para esse cenário.
<b>Fábrica de Software</b>	Sem comentário adicional para esse cenário.

### Orientadores de riscos (*Risk Driver*):

Riscos de não realizar essa tarefa:

- As necessidades das partes interessadas não serem atendidas, de forma eficaz e eficiente, pelo sistema, devido à não identificação de um atributo crítico de qualidade do sistema esperado pelas partes interessadas.
- Arquitetura do sistema inadequada.
- Retrabalho para ajustar da arquitetura do sistema para atendimento de requisitos não funcionais críticos do sistema.
- Infraestrutura de produção inapropriada para suportar a operação do sistema devido à não identificação de um atributo crítico de qualidade do sistema que exige capacidades e configurações específicas dos recursos computacionais da infraestrutura de TI do TSE.

### Informações adicionais:

- Esses tipos de requisitos são muito importantes para determinar a complexidade e o tamanho do sistema.
- Essas restrições são muito importantes para determinar a arquitetura adequada do sistema para atender às necessidades das partes interessadas.

### 3. Tabela RACI

Tarefas	Papéis									
	Dono do Produto	Partes Interessadas	Gerente de Projeto	Analista de Sistemas	Arquiteto de Software	Desenvolvedor	Analista de Teste	Testador	Administrador de Dados (Sead)	Administrador de infraestrutura de TI
1.1. Desenvolver a Visão do Sistema	C	C	A	R	C				C	
1.2. Localizar Atores e Casos de Uso	C	C	C	R/A	I		I		I	
1.3. Desenvolver Especificações Suplementares	C	C	I	R/A	C		I		I	
1.4. Captar um Vocabulário Comum	C	C		R/A					C	
1.5. Gerenciar Dependências de Requisitos			I	R/A	I				I	
2.1. Especificar Caso de Uso	C	C	I	R/A	I		I		C	
2.2. Elaborar Interface e Protótipo de Interface de Caso de Uso	C	C	I	R/A	C		I			
2.3. Refinar Especificações Suplementares	C	C	I	R/A	C		I		I	

Uma tabela RACI identifica quem é responsável (R), responsabilizado (A), consultado (C) e/ou informado (I).

\* Nota de tradução: (R) *Responsible*, (A) *Accountable*

#### 4. Produtos de Trabalho (Artefatos) da Disciplina

Segue lista dos artefatos da disciplina.

#	Artefato	Breve Descrição
01	<b>Documento de Visão do Sistema</b>	Esse artefato define a visão dos principais envolvidos ( <i>stakeholders</i> ) a respeito do produto a ser desenvolvido, especificada em termos das principais necessidades de negócio e recursos/características do sistema mais importantes. Ele contém uma breve descrição dos requisitos centrais pretendidos e, portanto, proporciona a base para requisitos técnicos mais detalhados.
02	<b>Especificação de Caso de Uso</b>	Especificação textual que contém uma descrição do fluxo de eventos que descreve a interação entre os atores e o sistema. A especificação também contém normalmente outras informações, como precondições, pós-condições, requisitos especiais e cenários principais.
03	<b>Especificação de <i>Layout</i> de Arquivos</b>	Esse documento pode ser utilizado de forma complementar à especificação de caso de uso, para o detalhamento de algum arquivo envolvido no processamento dos fluxos de eventos.
04	<b>Especificação de Perfis de Acesso</b>	Pode ser utilizado de forma complementar à especificação de caso de uso, para o detalhamento dos perfis de acesso dos atores em relação às funcionalidades do sistema.
05	<b>Especificação de Processamento de Dados</b>	Esse documento tem como finalidade descrever os dados, requisitos e regras aplicados no processamento de dados do sistema, podendo ser aplicado isoladamente ou de forma complementar à especificação de caso de uso.
06	<b>Especificação de Regras de Negócio</b>	Especificação textual das regras de negócios que descrevem as políticas, condições e restrições que devem ser satisfeitas pelo sistema.
07	<b>Especificação de Requisitos de <i>Software</i></b>	Esse artefato concentra requisitos de <i>software</i> para um sistema como um todo ou para uma parte dele, além de auxiliar a coleta e a organização do produto a ser construído ou evoluído, pode funcionar como um repositório em um documento formal.
08	<b>Especificação Suplementar</b>	Esse artefato captura os requisitos do sistema que não são prontamente capturados nos artefatos de requisitos comportamentais, como as especificações de casos de uso.
09	<b>Glossário do Sistema</b>	Esse artefato define termos importantes utilizados pelo domínio de negócio, sistema ou projeto. Termos identificados durante as atividades de requisitos, análise e <i>design</i> de sistema.
10	<b>Interface de Caso de Uso</b>	Esse documento descreve os elementos da interface do caso de uso. Seu escopo abrange a descrição das características dos campos de formulários que deverão compor cada interface do usuário relativa ao caso de uso em questão. Se esse caso de uso for responsável pela emissão de relatórios, a formatação e as informações que compõem os respectivos relatórios também estarão especificadas nesse documento.
11	<b>Mapeamento de Interdependências</b>	Documento da Metodologia de Gerenciamento de Projetos (MGP) para o mapeamento de interdependências, sendo, basicamente, uma forma de vincular entregas entre projetos.

## Guia de Referência Disciplina Requisitos

#	Artefato	Breve Descrição
12	<b>Matriz de Dependência entre Elementos</b>	Demonstra a relação de dependência entre elementos fonte e entidades. Esse modelo tem como objetivo proporcionar um maior controle sobre os elementos fonte gerados ao longo do processo de desenvolvimento, facilitando a análise de impacto quando houver uma solicitação de manutenção às entidades.
13	<b>Matriz de Rastreabilidade de Requisitos</b>	Uma tabela que liga os requisitos do sistema desde as suas origens até os componentes do sistema que os satisfazem.
14	<b>Mensagem de Sistema</b>	Esse documento descreve todas as mensagens de interface do usuário que serão referenciadas em mais de um caso de uso e que devem ser apresentadas aos usuários do sistema.
15	<b>Modelo de Análise (Domínio)</b>	O Modelo de Análise (Domínio) representa os conceitos, os produtos e os eventos que são importantes para o domínio de negócios. Esse modelo descreve apenas as informações significativas para o negócio e não inclui as responsabilidades que as pessoas têm.
16	<b>Modelo de Caso de Uso</b>	Modelo das funções pretendidas do sistema e de seu ambiente. Utilizado como um contrato de escopo do sistema estabelecido entre o cliente e os desenvolvedores. É utilizado como fonte de informações essencial para atividades de análise, projeto ( <i>design</i> ), medição, teste e gerenciamento do projeto.
17	<b>Protótipo de Interface do Usuário</b>	Exemplo da interface do sistema com o usuário. É utilizado para explorar e/ou validar os requisitos do sistema e o projeto ( <i>design</i> ) da interface com o usuário.
18	<b>Termo de Abertura de Projeto / Termo Simplificado de Abertura e Planejamento de Projeto / Plano de Ação (MGP/STI)</b>	Documentos da Metodologia de Gerenciamento de Projetos (MGP) da STI apreciados pela Comissão Técnica de Tecnologia da Informação (CTTI) que autoriza formalmente a existência de um projeto na STI/TSE e concede ao gerente do projeto a autoridade para aplicar os recursos organizacionais nas atividades do projeto.

## 4.1 Tabela de Entrada/Saída

### Artefatos

Tarefas	Glossário do Sistema	Modelo de Análise	Especificação de Requisitos de Software	Documento de Visão do Sistema	Matriz de Rastreabilidade de Requisitos	Matriz de Dependência entre Elementos	Modelo de Caso de Uso	Especificação Suplementar	Especificação de Caso de Uso	Especificação de Layout de Arquivos	Especificação de Processamento de Dados	Interface de Caso de Uso	Especificação de Regras de Negócio	Protótipo de Interface do Usuário	Mensagem de Sistema	Termo de Abertura de Projeto / Termo Simplificado de Abertura e Planejamento de Projeto (MGP/STI)
1.1. Desenvolver a Visão do Sistema			S	S												E
1.2. Localizar Atores e Casos de Uso			E	E			S									
1.3. Desenvolver Especificações Suplementares				E				S								
1.4. Captar um Vocabulário Comum	ES	ES		E			E	E	E				E			E
1.5. Gerenciar Dependências de Requisitos				E	ES	ES	E	E								E
3.1. Especificar Caso de Uso				E			E		S	S	S		S		S	
3.2. Elaborar Interface e Protótipo de Interface de Caso de Uso									E			S		S		
3.3. Refinar Especificações Suplementares				E				ES								

Uma tabela identifica o que é usado (E – Entrada), modificado (ES – Entrada e Saída) ou produzido (S-Saída) por uma atividade ou tarefa.



## 5. Onde Posso Aprender Mais sobre a Disciplina?

PRESSMAN, Roger S. **Engenharia de software**. 6. ed. Rio de Janeiro: McGraw-Hill, 2006.

SOMMERVILLE, Ian. **Engenharia de software**. 8. ed. São Paulo: Addison-Wesley, 2007.

SOFTWARE ENGINEERING INSTITUTE. **CMMI for Development (CMMI-DEV)**, Version 1.3, Technical Report CMU/SEI-2010-TR-033. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2010

ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX. **MPS.BR – Guia Geral**:2011, junho 2011. Disponível em [www.softex.br](http://www.softex.br)

CMUSEI, Carnegie Mellon University, Software Engineering Institute. **CMMI for Development Version 1.3**. Pensilvania: Carnegie Mellon University, 2010.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML Guia do Usuário**. 2ª ed. Rio de Janeiro: Campus, 2006.

IBM. **Rational Unified Process®**, Versão 7.0.1, IBM Corporation, Copyright © 1987 – 2006.

BITTNER, K; SPENCE, I. **Use Case Modeling**. Nova Iorque: Addison Wesley, 2002.

COCKBURN, Alistair. **Escrevendo Casos de Uso Eficazes**. [s.l.]: Bookman, 2005.

ROSENBERG, D; SCOTT, K. **Applying Use Case Driven Object Modeling with UML – An annotated e-Commerce Example**. [s.l.]: Addison Wesley, 2001.

ROSENBERG, D; SCOTT, K. **Use Case Driven Object Modeling with UML – A practical Approach**. [s.l.]: Addison Wesley, 2003.

WINDLE, D; ABREO, L. **Software Requirements Using the Unified Process: A Practical Approach**. [s.l.]: Prentice Hall, 2002.

ISO – International Organization for Standardization / IEC – International Electrotechnical Commission  
ISO/IEC (2006) " **ISO/IEC 14764: Software Engineering – Software Life Cycle Processes – Maintenance**," International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC), ISO/IEC International Standard.

## Guia de Referência Disciplina Requisitos

### 6. Apêndice I – Tabela Relacionando Tarefas e Objetivos da Disciplina

Tarefas	Objetivo da Disciplina						
	Concordância com as partes interessadas	Limites do sistema	Compreensão dos requisitos do sistema	Planejamento técnico e gerencial	Tamanho funcional do sistema	Custo e tempo de desenvolvimento	Interface de usuário do sistema
1.1. Desenvolver a Visão do Sistema	✓	✓	✓	✓	✓	✓	
1.2. Localizar Atores e Casos de Uso		✓	✓	✓	✓	✓	
1.3. Desenvolver Especificações Suplementares			✓	✓	✓	✓	✓
1.4. Captar um Vocabulário Comum			✓		✓		
1.5. Gerenciar Dependências de Requisitos		✓	✓	✓		✓	
2.1. Especificar Caso de Uso			✓	✓	✓	✓	✓
2.2. Elaborar Interface e Protótipo de Interface de Caso de Uso					✓		✓
2.3. Refinar Especificações Suplementares			✓	✓	✓	✓	✓