

Nome do Sistema
Especificação de Objetivos e Requisitos

SIGLASISTEMA-EOR-999

Local e data

EOR	Versão:
Nome do Software	Data/Hora:

Revisões

A primeira versão deste documento é criada após a sua aprovação e é vinculada a uma baseline do software. Esta versão, portanto, não pode ser modificada. As modificações que se fizerem necessárias após a criação da baseline farão parte de uma versão seguinte que será vinculada a outra baseline. Este procedimento pode se repetir sucessivamente. As modificações introduzidas em cada versão devem ser registradas seguindo o modelo do quadro abaixo. Desta forma, será possível perceber as diferenças entre as diversas versões geradas.

<i>Data</i>	<i>Descrição</i>	<i>Autor</i>
Incluir a data da modificação	Descrever resumidamente o motivo da revisão	Informar o nome do responsável pela modificação

Conteúdo

Incluir índice de conteúdo do documento que estiver sendo elaborado, conforme exemplo abaixo usado para esta proposta de metadocumento:

1.	Introdução	3
1.1.	Objetivos.....	3
1.2.	Público Alvo	4
1.3.	Organização do documento	4
2.	Descrição do problema e do sistema	4
2.1.	Identificação e missão do Sistema.....	4
2.2.	Domínio do problema e contexto de sua aplicação	4
2.3.	Objetivos e Benefícios Esperados do Sistema.....	4
2.4.	Características Essenciais do Sistema	5
2.5.	Descrição dos interessados do sistema	5
2.6.	Diagnostico da Situação Atual	5
2.6.1	Áreas de negócio atendidas	5
2.6.2	Áreas de negócio não atendidas.....	5
2.6.3	Integração com outros sistemas da empresa.....	5
2.6.4	Integração com outros sistemas de terceiros	6
2.6.5	Pontos negativos	6
2.6.6	Pontos positivos.....	6
2.6.7	Sugestões de melhorias.....	6
3.	Casos de Uso	6
3.1.	Atores	6

EOR	Versão:
Nome do Software	Data/Hora:

3.2.	Diagramas de Caso de Uso e Lista de casos de uso	6
3.3.	Descrição de Casos de Uso.....	7
4.	Requisitos e restrições funcionais (RFUN)	7
5.	Requisitos e restrições não funcionais.....	7
5.1.	Requisitos e restrições de informação (RINF)	7
5.2	Requisitos e restrições de interface Homem-Computador (RHIC).....	8
5.3	Requisitos de Interface Externa (RIEX).....	9
5.4	Requisitos e Restrições de Projeto (RPRO)	9
5.5	Requisitos e restrições de arquitetura de software (RARQ)	10
5.6	Requisitos e restrições de plataforma de hardware (RPHW).....	10
5.7	Requisitos e restrições de plataforma de software (RPSW).....	10
5.8	Requisitos e restrições de desempenho (RDES)	11
5.9	Requisitos e restrições de disponibilidade (RDIS).....	11
5.10	Requisitos e restrições de segurança (RSEG)	11
5.11	Requisitos e restrições de manutenibilidade (RMAN).....	12
5.12	Requisitos e restrições de portabilidade (RPOR).....	12
5.13	Requisitos de documentação (RDOC)	13
6.	Requisitos Futuros (RFUT)	13
7.	Referências cruzadas complementares	13
8.	Aprovação Formal	14
9.	Bibliografia.....	14
Anexos		15

Figuras e Tabelas

Incluir índice de figuras e tabelas mostradas no documento usando o mesmo modelo do índice de conteúdo.

1. Introdução

Apresentar o documento ao leitor, descrevendo sucintamente o software que é objeto deste projeto e as informações contidas neste documento.

1.1. Objetivos

Definir os objetivos do documento.

Exemplo:

EOR	Versão:
Nome do Software	Data/Hora:

Este documento descreve todos os casos de uso, requisitos que deverão ser atendidos pelo Sistema XXXX que será desenvolvido para a empresa XXXXXXXX.

1.2. Público Alvo

Identificar o público alvo do documento, isto é, os órgãos e pessoas que poderão usá-lo.

1.3. Organização do documento

Descrever a organização e o conteúdo de cada seção do documento.

2. Descrição do problema e do sistema

Nesta seção o sistema objeto da especificação de requisitos deve ser descrito. O domínio do problema que deverá ser resolvido pelo sistema deve ser explicado e também as características específicas do problema no contexto da empresa em que o sistema deverá ser utilizado. As subseções seguintes podem ser colocadas no documento EOR para separar cada conteúdo específico, mas também pode ser usado um estilo de texto sem separação de subseções desde que todo o conteúdo seja coberto.

2.1. Identificação e missão do Sistema

Identificar o sistema objeto da especificação de requisitos e definir em poucas palavras qual é a missão do sistema que está sendo considerado neste documento.

2.2. Domínio do problema e contexto de sua aplicação

A descrição do domínio do problema deve complementar e detalhar a visão geral do domínio do problema apresentada no documento de ativação do projeto (APD), identificando o contexto deste problema no ambiente alvo (ambiente do cliente alvo).

Exemplo:

Suponha que o domínio do problema seja a venda de produtos em um comércio varejista. Esta seção deverá explicar em que consiste o problema de vender produtos em uma empresa que atua no comércio varejista, ou seja, descrever genericamente quais são os processos de negócio envolvidos e seus objetivos. Além disso, o problema deve ser contextualizado mostrando as características específicas do problema na empresa alvo.

2.3. Objetivos e Benefícios Esperados do Sistema

Esta seção deve refinar a definição do produto contida no documento APD, ajudando a estabelecer o escopo do sistema a ser desenvolvido. Listar todos os objetivos a serem alcançados com o desenvolvimento do sistema e os respectivos benefícios esperados.

Exemplo:

<i>Objetivo a ser alcançado</i>	<i>Benefício adquirido</i>
<i>Atualizar automaticamente o estoque de produtos após a realização de uma venda.</i>	<i>Permitir que os vendedores tenham informações atualizadas sobre os produtos disponíveis em estoque;</i>

EOR	Versão:
Nome do Software	Data/Hora:

2.4. Características Essenciais do Sistema

Descrever todas as características que devem ser consideradas para avaliar o sucesso do produto construído. Os detalhes necessários a essa avaliação devem ser especificados. Por exemplo, se a capacidade do sistema ser executado em diferentes plataformas computacionais é especificada como uma característica do sistema, deve ser definido um conjunto de plataformas que serão usadas para avaliar essa característica.

O produto final do processo de desenvolvimento será validado com relação à missão, aos objetivos e às características descritas neste documento.

2.5. Descrição dos interessados do sistema

Descrever os perfis de cada interessado envolvido com o sistema.

<i>Interessado(s)</i>	<i>Descrição</i>	<i>Necessidades Básicas</i>
Nome do interessado. Por exemplo: (cliente, atendente, Almoxarifado, etc.)	Descrever o interessado e sua interação com o sistema	Indicar quais as necessidades básicas que o interessado tem em relação ao sistema. Por exemplo: o vendedor precisa de um sistema que lhe permita registrar uma venda feita com segurança e rapidez.

2.6. Diagnostico da Situação Atual

Descrever a situação atual do sistema que será substituído e/ou informatizado. O software a ser desenvolvido pode substituir outro software, complementar as funcionalidades de outro software ou pode servir para informatizar um sistema manual. Em ambos os casos, é preciso analisar o sistema que será modificado pela implantação do novo software.

2.6.1 Áreas de negócio atendidas

Descrever as áreas de negócio (unidades organizacionais) dos clientes (ou da própria empresa, no caso de sistemas de uso interno), que utilizam o sistema atual. Para cada área de negócio identificada deverão ser indicados quais os processos de negócio realizados e como cada processo é executado (para quê, quando e como). Além disso, devem ser identificadas as informações manipuladas no processo.

2.6.2 Áreas de negócio não atendidas

Descrever os processos de negócio que deveriam ou poderiam ser realizados no sistema atual, mas não o são. Identificar que tipo de suporte deveria ser fornecido pelo sistema (o quê deveria ser feito, quando e como) e que tipo de informações seriam necessárias para esse fim.

2.6.3 Integração com outros sistemas da empresa

Explicar como o sistema atual se integra com outros sistemas da empresa (que tipo de integração existe, como, quando e por que ela é feita).

EOR	Versão:
Nome do Software	Data/Hora:

2.6.4 Integração com outros sistemas de terceiros

Explicar como o sistema atual se integra com sistemas desenvolvidos por outras empresas (que tipo de integração existe, como, quando e por que ela é feita).

2.6.5 Pontos negativos

Identificar os problemas e dificuldades que o sistema atual apresenta em termos de funcionalidades, desempenho ou qualquer outro aspecto relevante.

2.6.6 Pontos positivos

Identificar as virtudes e facilidades que o sistema atual apresenta em termos de funcionalidades, desempenho ou qualquer outro aspecto relevante.

2.6.7 Sugestões de melhorias

Listar as principais sugestões de melhorias para o sistema atual, que foram coletadas com clientes, usuários e colaboradores da empresa. Essas sugestões deverão ser classificadas em níveis de prioridade (por exemplo: alta, média, baixa, e descartada).

3. Casos de Uso

Nesta seção devem ser identificados os casos de uso do software que será desenvolvido, os atores neles envolvidos e os requisitos funcionais e não funcionais neles tratados.

3.1. Atores

Listar todos os atores do software. Ator é tudo aquilo que interage com o software, por exemplo, pessoas, órgãos, software, máquinas, etc.

Exemplo:

1. Cliente
2. Software de Faturamento
3. Atendente
4. Telefonista
5. Supervisor

3.2. Diagramas de Caso de Uso e Lista de casos de uso

Mostrar os casos de uso do sistema através de diagrama(s) de caso de uso identificados. Listar todos os casos de uso do software identificando sua categoria: primário, secundário ou opcional. Casos de uso primários são aqueles que representam processos comuns principais; casos de uso secundários representam processos menos importantes ou raros; casos de uso opcionais representam processos que talvez não sejam considerados. Quando define a categoria de um caso de uso é mais fácil de perceber quais casos de uso deverão ser expandidos primeiramente.

Exemplo:

EOR	Versão:
Nome do Software	Data/Hora:

Ref.	Descrição	Atores	Categoria
CSU1	Registrar ligação	Telefonista	Primário
CSU2	Solicitar impressão de dados de clientes	Atendente	Secundário

3.3. Descrição de Casos de Uso

Identificar e descrever cada caso de uso listado anteriormente.

Os casos de uso deverão ser descritos preferencialmente no formato de alto nível, mas dependendo da sua importância ou da sua complexidade eles poderão ser descritos no formato essencial expandido. Casos de uso essenciais são aqueles que não referenciam aspectos de soluções tecnológicas adotadas ao contrário dos casos de uso em formato concreto que referenciam as estruturas de interface da solução projetada para o sistema e outros aspectos tecnológicos do sistema.

4. Requisitos e restrições funcionais (RFUN)

Elaborar uma lista das funções que o software deve prover. Considerar não apenas as funções intrínsecas (essenciais) do software, mas também as de suporte, tais como administração da utilização do software, cópias de segurança, auditoria, e controle de acesso.

Indicar para cada função uma referência, descrição, categoria, prioridade e casos de uso em que ela é tratada.

Exemplo:

Ref.	Função	Categoria	Prioridade	Casos de Uso
RFUN1	Registrar venda de um produto	Evidente	Alta	CSU _x , CSU _y
RFUN2	Atualizar o estoque do produto vendido	Oculto	Média	CSU _z

5. Requisitos e restrições não funcionais

Elaborar uma lista de todos os requisitos não funcionais. Considerar requisitos de informação, de interface, de projeto, de arquitetura de software, de plataforma de hardware, de plataforma de software, de plataforma de comunicação, de desempenho, de disponibilidade, de segurança, de manutenibilidade, de portabilidade e de documentação. A lista poderá ser dividida por tipo de requisito, mas é importante que os requisitos tenham uma identificação única para que possam ser referenciados sem ambigüidades no futuro.

5.1. Requisitos e restrições de informação (RINF)

Elaborar uma lista de todas as necessidades de informação que o software não pode deixar de atender. Esta lista deverá ser classificada em informações cadastrais e

EOR	Versão:
Nome do Software	Data/Hora:

informações gerenciais. Por exemplo, para um software de vendas existem, entre outras, as seguintes necessidades de informação:

Exemplo:

Ref.	Tipo	Descrição	Casos de Uso
RINF1	Cadastral	Vendedor: matrícula, nome, documentos de identificação, salário, endereço e telefone.	CSUx
RINF2	Cadastral	Produto: código, descrição, unidade de armazenamento, preço.	CUSy, CUSz
RINF3	Gerencial	Relação dos dez vendedores que mais venderam um determinado produto em um determinado período, contendo a matrícula e o nome do vendedor, o nome do produto, a quantidade vendida, o valor total da venda e a comissão paga.	CSUw

Estes requisitos de informação são importantes para verificar a qualidade da modelagem de dados que for feita.

5.2 Requisitos e restrições de interface Homem-Computador (RHIC)

Definir todos os aspectos de Interface Homem Computador (IHC) incluindo: conteúdo de informações, fatores ergonômicos, dispositivos de interação, formato de apresentação, tipo de diálogo, e mecanismos de ajuda alocados a cada perfil/grupo/tarefa de usuário. Descrever, em particular, os requisitos de usabilidade para cada perfil/grupo/tarefa de usuário. Por exemplo, pode-se definir como requisito que as opções de *menu* do sistema tenham teclas de atalho associadas..

É recomendável definir diagramas de interface (telas e relatórios) para as funções previstas para o software. Devem ser estabelecidas, no diagrama, as áreas da janela ou do relatório destinadas a cada tipo de informação.

Existem diversos padrões para definição de interface de usuário. Exemplos destes padrões predefinidos são *Motif* e *Windows*. Esses padrões de interface definem desde diretivas para diagramação até os tipos de objetos de interface que podem ser adotados, com seus respectivos atributos e valores default.

No caso de se adotar algum padrão predefinido de interface, pode-se apenas referenciá-lo nesta seção, já que a diagramação de todas as janelas e relatórios deve seguir este padrão. Caso contrário, para cada área funcional identificada na diagramação, deve-se especificar a sua finalidade, suas dimensões e seu posicionamento relativo na janela ou relatório. Restrições sobre o tamanho e posição relativos entre as diversas áreas do diagrama também devem ser especificadas.

Cada área do diagrama pode ser recursivamente subdivida em áreas menores. Nestes casos devem ser especificadas as mesmas informações definidas para as áreas principais, ou seja, a identificação, a finalidade, o tamanho e o posicionamento relativo.

EOR	Versão:
Nome do Software	Data/Hora:

Exemplos típicos de áreas funcionais de janelas incluem: área de mensagens; área de comandos; área de respostas a comandos; área de desenho; área de menus; e área de identificação da Janela. As áreas típicas de composição de um relatório são: cabeçalhos, corpo, linhas de detalhe, linhas de totalização, e rodapé.

Exemplo:

Ref.	Descrição	Casos de Uso
RIHC1	Para facilitar a usabilidade na transação de venda, pede-se que a tela de vendas tenha uma fonte (tipo e tamanho de letra) que permita uma fácil visualização a uma distância de 2 metros do monitor porque desta forma o cliente poderá visualizar as informações da venda da sua posição.	Todos

5.3 Requisitos de Interface Externa (RIEX)

Identificar e descrever as interfaces com outros softwares/sistemas que o software deverá prover. Por exemplo, um software comercial deve gerar informações para o Sistema de Arrecadação da Secretaria da Fazenda Estadual. O formato dessas informações e o protocolo de envio são definidos pela própria secretaria, e atender essas definições é um requisito do software.

Exemplo:

Ref.	Descrição	Casos de Uso
RIEX1	O software do sistema de vendas deverá gerar um arquivo SINTEGRA conforme Legislação do Convênio ICMS 57/95 atualizado até 69/02, incluindo as alterações posteriores, para que seja enviado para a Secretaria da Fazenda trimestralmente.	CSUn

5.4 Requisitos e Restrições de Projeto (RPRO)

Nesta seção serão especificados todos os requisitos e restrições associados a condução do projeto de desenvolvimento e que podem limitar ou definir ações que serão executadas.

Exemplo:

Ref.	Descrição	Casos de Uso
RPRO1	O cliente solicitou que o módulo de contabilidade fosse entregue até o dia 10 do último mês do ano corrente (10/12/05), para testes em ambiente real.	CSUi, CSUj, CSUk

EOR	Versão:
Nome do Software	Data/Hora:

5.5 Requisitos e restrições de arquitetura de software (RARQ)

Se o software tiver de ser desenvolvido em uma arquitetura específica, então essa arquitetura deverá ser descrita.

Exemplo:

Ref.	Descrição	Casos de Uso
RARQ1	O software deverá ser desenvolvido com uma arquitetura de camadas que permita isolar as funcionalidades ligadas ao negócio das funcionalidades relacionadas com a interface homem-computador	Todos

5.6 Requisitos e restrições de plataforma de hardware (RPHW)

Identificar e descrever requisitos e restrições relacionadas com a plataforma de hardware que será utilizada pelo software:

Exemplo:

Ref.	Descrição	Casos de Uso
RPHW1	O software deverá ser capaz de rodar em um Servidor com processador Intel xSeries (IBM)	

5.7 Requisitos e restrições de plataforma de software (RPSW)

Se o software tiver que ser executado em plataformas de software específicas, essas plataformas de software deverão ser definidas:

- Sistema Operacional: identificar e descrever o sistema operacional em que o software deverá ser executado;
- Softwares Básicos: identificar SGBD, linguagem de programação, ferramentas CASE e outros.

Se houver mais de uma plataforma de software, deve-se especificar qual a plataforma principal e em que situações as outras plataformas podem ser utilizadas.

Exemplo:

Ref.	Descrição	Casos de Uso
RPSW1	O software deverá ser desenvolvido com a ferramenta CASE XYZ gerando código Java. A justificativa para esta restrição é que esta é plataforma-padrão adotada pela empresa.	Todos

EOR	Versão:
Nome do Software	Data/Hora:

5.8 Requisitos e restrições de desempenho (RDES)

Identificar e descrever os requisitos e restrições de desempenho do software.

Exemplo:

Ref.	Descrição	Casos de Uso
RDES1	O ambiente onde o software rodará deverá permitir pelo menos três usuários acessando o banco de dados sem queda de velocidade.	Todos
RDES2	O tempo de resposta máximo permitido para transações on-line é de 5 segundos	CSUx,...
RDES3	O software deverá ser capaz de atender até dez transações simultâneas da função “Registrar Venda”.	CSUz

5.9 Requisitos e restrições de disponibilidade (RDIS)

Especificar os requisitos de disponibilidade necessários para o software de uma forma global:

- Período de disponibilidade: horário comercial, 24 horas por dia, etc.
- Período máximo para recuperação do software em caso de falha.

Devem ser definidos os tipos de falha e a tolerância aceitável para cada tipo de falha. Os tipos de falha podem ser definidos em função dos requisitos funcionais e de dados, mas não se restringem a estes. Por exemplo: a função “Registrar Venda” deve ter um tempo para recuperação de falha de no máximo uma hora (o que significa que esta função não poderá ficar mais do que uma hora indisponível para o usuário em nenhuma circunstância).

Exemplo:

Ref.	Descrição	Casos de Uso
RDIS1	O software deverá estar disponível 24 horas por dia	Todos

5.10 Requisitos e restrições de segurança (RSEG)

Especificar os requisitos de segurança necessários para controle de acesso ao software. Definir a necessidade de:

- Verificação de senha;
- Criptografia de dados;
- Registro das operações efetuadas;
- Habilitação de funções por perfil de usuário;
- Acesso seletivo aos dados e funções.

EOR	Versão:
Nome do Software	Data/Hora:

Exemplo:

Ref.	Descrição	Casos de Uso
RSEG1	O software deverá solicitar autorização do supervisor para excluir uma venda já registrada.	CSUv
RSEG2	Todos os dados em arquivos de exportação deverão ser criptografados.	CSUx
RSEG3	Para qualquer atualização efetuada, deverá ser registrado o usuário que realizou a operação, bem como a data e hora.	CSUn, CSUm,...
RSEG4	Todo usuário do software deverá ser associado a um perfil que define as funcionalidades que poderão ser utilizadas por ele.	Todos

5.11 Requisitos e restrições de manutenibilidade (RMAN)

Especificar os requisitos que visam facilitar a manutenção posterior do software, tais como:

- Requisitos de reutilização (exemplo: uso de implementação orientada a objetos; bibliotecas de classes e padrões de projeto);
- Requisitos de modularização (exemplo: valores para métricas de acoplamento entre módulos; máximo de pontos de função por módulo);
- Requisitos de configuração (exemplo: regras para controle de versões);
- Requisitos de documentação (exemplo: documentação de programa)

Exemplo:

Ref.	Descrição	Casos de Uso
RMAN1	O projeto das responsabilidades de cada classe de objetos deverá seguir os padrões GRASP sugeridas no livro Utilizando UML e Padrões de Craig Larman [1].	Todos
RMAN2	Todo programa deve estar documentado de acordo com as orientações contidas na Norma de Documentação de Programas da empresa [2]	Todos

5.12 Requisitos e restrições de portabilidade (RPOR)

Identificar as diversas plataformas de software e hardware com as quais o software deve ser compatível. Devem ser consideradas tanto plataformas de desenvolvimento como plataformas de produção. Outros exemplos de requisitos de portabilidade são:

- Percentual de componentes que podem ter código dependente da plataforma hospedeira;
- Percentual de código que pode ser dependente da plataforma hospedeira;
- Uso de uma linguagem reconhecidamente portátil.

EOR	Versão:
Nome do Software	Data/Hora:

Exemplo:

Ref.	Descrição	Casos de Uso
RPOR1	O software deverá ser capaz de rodar tanto em computadores Desktop com Windows 98 ou superior ou com Linux. Deverá ser capaz de rodar também em Palmtops com Windows CE.	CSUx, CSUy
RPOR2	O software deverá permitir a utilização dos seguintes Bancos de dados: SQL Server, MySql ou Oracle	Todos

5.13 Requisitos de documentação (RDOC)

Especificar os requisitos de documentação do produto de software que será desenvolvido.

Exemplo:

Ref.	Descrição	Casos de Uso
RDOC1	Junto com o script de instalação do produto deve ser colocado um manual de instalação do mesmo	Não se aplica
RDOC2	O software deve prover ajuda on-line sensível ao contexto em todas as suas telas	Todos

6. Requisitos Futuros (RFUT)

Descrever os requisitos que poderão ser especificados em uma nova versão do produto.

Exemplo:

Ref.	Descrição	Casos de Uso
RFU1	Em um futuro próximo o software de atendimento de clientes deverá ser integrado com o software do sistema de faturamento para que o atendente possa identificar o perfil de negócios do cliente	CSUx

7. Referências cruzadas complementares

Nesta seção são colocadas algumas referências cruzadas que podem ajudar o rastreamento futuro dos requisitos. Estes mapeamentos podem ser feitos em forma de matrizes de rastreabilidade como mostram os exemplos a seguir:

EOR	Versão:
Nome do Software	Data/Hora:

Requisitos Funcionais	Requisitos não funcionais vinculados
<i>Colocar a identificação do requisito funcional</i>	<i>Colocar a identificação do requisito não funcional vinculado</i>

Caso de Uso	Requisitos funcionais vinculados
<i>Colocar a identificação do caso de uso</i>	<i>Colocar a identificação do requisito vinculado</i>

Requisito Funcional	Origem do Requisito
<i>Colocar a identificação do requisito funcional</i>	<i>Colocar a origem do requisito. Pode ser uma entrevista, um questionário, ou outra técnica qualquer de elicitação de requisito aplicada.</i>

Requisito Não Funcional	Origem do Requisito
<i>Colocar a identificação do requisito não funcional</i>	<i>Colocar a origem do requisito. Pode ser uma entrevista, um questionário, ou outra técnica qualquer de elicitação de requisito aplicada.</i>

Caso de Uso	Origem do caso de uso
<i>Colocar a identificação do requisito funcional</i>	<i>Colocar a origem do caso de uso. Pode ser uma entrevista, um questionário, ou outra técnica qualquer de elicitação de requisito aplicada.</i>

Observações:

1. Os exemplos acima não esgotam as possibilidades de referências cruzadas.
2. Sugere-se que as referências sejam feitas usando *hiperlinks* sempre que possível.

8. Aprovação Formal

O EOR deve ser datado e assinado de acordo com a definição contida no MPD. O documento aprovado se torna uma *baseline* do projeto que passa a ser controlada pela Gerência de Configuração de Software.

9. Bibliografia

Detalhar todas as fontes de informação citadas no documento ou usadas de alguma forma para sua elaboração, incluindo título, autor, data, fonte de obtenção, órgão

EOR	Versão:
Nome do Software	Data/Hora:

responsável pela publicação e qualquer outra informação que possa facilitar a localização e obtenção dessas fontes pelo leitor. Nesta seção cada fonte citada deve ser associada a um número identificador. Junto de cada citação deve ser colocada também uma explicação sucinta de como a fonte foi usada na elaboração do documento.

Referenciar todas as fontes de informações citadas no EOR, explicando o motivo de sua citação. Devem ser incluídos, por exemplo:

- Normas e Padrões utilizados;
- Livros e artigos (autores, título, editora, páginas, volume, data e local da publicação, ISBN/ISSN, etc);
- Manuais de outros Produtos.

Por exemplo, vamos supor que para elaborar o documento XXX-EOR-001 foram usadas as orientações contidas no Meta-Modelo de Especificação de Objetivos e Requisitos (EOR). Então, nesta seção deveria aparecer o seguinte texto:

*[1] Ronaldo Lopes de Oliveira, **EOR – Modelo de Documento de Especificação de Objetivos e Requisitos de Software, Versão 1.0, agosto de 2005.***

Meta-Modelo usado como guia para definição do formato e conteúdo deste documento.

10. Anexos

Anexar ao EOR documentos considerados importantes para sua compreensão. Por exemplo, documentos e leis usadas na especificação dos requisitos podem ser colocados nos anexos para facilitar o acesso do leitor às fontes dos requisitos do software. Cada anexo deverá receber uma folha de rosto indicando seu conteúdo.