

<Nome do Sistema>

Especificação de Objetivos e Requisitos

<SIGLASISTEMA>-EOR-v001

<Local e data>

EOR	Versão:
Nome do Software	Data/Hora:

Conteúdo

Incluir índice de conteúdo do documento que estiver sendo elaborado

1.	Introdução	4
1.1.	Objetivos.....	4
1.2.	Público Alvo	4
1.3.	Organização do documento	4
2.	Descrição do problema e do sistema	5
2.1.	Identificação e missão do Sistema.....	5
2.2.	Domínio do problema e contexto de sua aplicação	5
2.3.	Descrição dos interessados do sistema	5
2.4.	Objetivos e Características Esperadas do Sistema	5
2.5.	Diagnóstico da Situação Atual	6
3.	Casos de Uso e Requisitos Funcionais	7
3.1.	Diagramas de Caso de Uso e Lista de casos de uso	7
3.2.	Descrição de Casos de Uso.....	7
4.	Requisitos e restrições não funcionais.....	8
4.1.	Requisitos de informação (RINF).....	8
4.2	Requisitos/Restrições de Interface Homem-Computador (RHIC)	9
4.3	Requisitos e Restrições de Interface Externa (RIEX)	9
4.4	Requisitos/Restrições de Projeto (RPRO)	10
4.5	Requisitos/Restrições de Arquitetura de software (RARQ)	10
4.6	Requisitos/Restrições de Plataforma de Hardware (RPHW)	10
4.7	Requisitos/Restrições de Plataforma de Software (RPSW)	11
4.8	Requisitos/Restrições de Desempenho (RDES)	11
4.9	Requisitos/restrições de disponibilidade (RDIS).....	11
4.10	Requisitos/Restrições de Segurança (RSEG)	12
4.11	Requisitos/Restrições de Manutenibilidade (RMAN).....	12
4.12	Requisitos/Restrições de Portabilidade (RPOR)	13
4.13	Requisitos/Restrições de Documentação (RDOC)	13
5.	Requisitos Futuros (RFUT)	14
6.	Referências cruzadas complementares	14
7.	Aprovação Formal	15
8.	Bibliografia.....	15

EOR	Versão:
Nome do Software	Data/Hora:

9. Anexos.....	15
----------------	----

Figuras e Tabelas

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

EOR	Versão:
Nome do Software	Data/Hora:

1. Introdução

Esta seção deverá apresentar o **documento** ao leitor. O objetivo principal da introdução é descrever o documento e não o sistema, embora deva ser mencionado qual é o sistema e para qual empresa ou mercado ele será construído. O conteúdo a ser coberto na introdução está descrito nas subseções seguintes.

1.1. Objetivos

Definir os objetivos do documento.

Exemplo:

Este documento tem os seguintes objetivos:

- *Definir os interessados no sistema e as suas necessidades que devem ser satisfeitas pelo sistema a ser desenvolvido;*
- *Derivar os casos de uso e requisitos do sistema de forma a orientar a equipe de que será responsável pelo seu desenvolvimento;*
- *Estabelecer um contrato para negociação e concordância entre todos os interessados;*
- *Reduzir retrabalho com projeto, codificação e teste através da especificação rigorosa e completa dos requisitos;*
- *Prover uma base para avaliação de prazos e custos de desenvolvimento;*
- *Facilitar a transferência dos produtos do desenvolvimento para novos usuários, novos clientes, novos ambientes operacionais e novas equipes de desenvolvimento e manutenção;*
- *Prover uma base para a evolução futura do sistema a partir de uma versão aprovada (linha de base) deste documento.*

1.2. Público Alvo

Identificar o público alvo do documento, isto é, todos os perfis de pessoas que terão interesse na sua leitura. Os interesses de cada perfil na leitura do documento também devem ser descritos.

1.3. Organização do documento

Descrever sucintamente a organização do documento em seções e o conteúdo de cada seção. Não há necessidade de descer ao nível de subseções.

EOR	Versão:
Nome do Software	Data/Hora:

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 ou mercado 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 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. Descrição dos interessados do sistema

Descrever os perfis de cada interessado envolvido com o sistema.

Interessado(s)	Descrição
Nome do perfil do interessado. Por exemplo: (cliente, atendente, vendedor, gerente comercial, etc.)	Descrever o perfil do interessado e como se espera que seja sua interação com o sistema direta ou indiretamente. Por exemplo, o vendedor da loja é a pessoa que atende os clientes e registra as vendas que são feitas.

Tabela 1 - Interessados no Sistema

2.4. Objetivos e Características Esperadas do Sistema

Esta seção deve listar, de acordo com a visão dos interessados no sistema, todos os objetivos e características fundamentais do sistema a ser desenvolvido e os respectivos benefícios esperados para o negócio ou para o interessado.

EOR	Versão:
Nome do Software	Data/Hora:

Exemplo:

<i>Interessado(s)</i>	<i>Objetivo ou característica desejada</i>	<i>Benefício adquirido</i>
<i>Vendedor, gerente de estoque, gerente de vendas</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 evitando vendas de produtos não disponíveis em estoque.</i>
<i>Gerente de vendas</i>	<i>O sistema deve ser utilizado por no mínimo cinco vendedores ao mesmo tempo com tempo de consulta a um produto não excedendo cinco segundos.</i>	<i>Permitir que vendas sejam feitas de forma ágil deixando os clientes satisfeitos</i>

Tabela 2 – Objetivos, características e benefícios esperados do Sistema

2.5. Diagnóstico da Situação Atual

Há três casos distintos a serem considerados nesta seção:

1. Sistema de software a ser desenvolvido para uma organização ou cliente específico para substituir um sistema já existente.
2. Sistema de software a ser desenvolvido para uma organização ou cliente específico, mas sem substituir um sistema já existente.
3. Sistema de software a ser desenvolvido sem ter cliente específico atendendo um determinado nicho de mercado como, por exemplo, softwares de prateleira, jogos, etc.

No primeiro caso deve ser descrita a situação do sistema existente na organização: pontos positivos e negativos; áreas do negócio que ele atende; áreas de negócio que não atende e que deveria atender; outros sistemas da empresa, próprios ou de terceiros, que ele se integra e como ocorre essa integração (quando, por que e como é feita); sugestões de melhorias dos usuários em relação ao sistema.

No segundo caso deve ser descrita a situação atual em relação a não existência de um sistema informatizado que atenda as necessidades da organização. Deve ser descrita a forma como o problema está sendo tratado na organização e quais as dificuldades e problemas decorrentes desse tratamento dado.

No terceiro caso deve ser descrita a realidade atual do mercado alvo do software em termos de potencial de aceitação e de rentabilidade (se for o caso) e de outros softwares existentes para o mesmo propósito ou propósito semelhante. Os softwares concorrentes deverão ser avaliados em relação aos seus pontos fortes e fracos e a possíveis diferenciais competitivos.

A seção pode ser subdividida em subseções caso seja julgado conveniente.

EOR	Versão:
Nome do Software	Data/Hora:

3. Casos de Uso e Requisitos Funcionais

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

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

Mostrar os casos de uso do sistema através de diagrama de caso de uso. Podem ser usados mais de um diagrama para facilitar a visualização e compreensão do leitor caso o número de casos de uso seja considerado grande.

3.2. Descrição de Casos de Uso

Cada caso de uso deve receber um identificador único, ser categorizado em primário ou secundário e ser descrito em um formato essencial resumido conforme explicado no capítulo 6 da terceira edição do livro “Utilizando UML e Padrões” de Craig Larman. Veja o exemplo a seguir:

CSU1- Processar Venda

Atores: Vendedor e Cliente

Categoria: Primário

Descrição:

Este caso de uso acontece quando um cliente chega a um ponto de venda com itens de produtos que deseja adquirir. O caixa usa o sistema para registrar cada item comprado. O sistema vai apresentando um total parcial e uma linha de detalhes à medida que registra cada item. O cliente fornece ao caixa os dados sobre o pagamento, que são então validados e registrados pelo sistema. O sistema atualiza o estoque e emite um recibo que é entregue pelo caixa ao cliente. O cliente sai com os itens comprados.

Requisitos Funcionais:

Ref.	Descrição	Categoria	Prioridade
RFUN 1.1	Obter e exibir a descrição e valor unitário de um produto a partir de seu código identificador	Evidente	Alta
RFUN 1.2	Calcular e exibir o total parcial da venda a cada item informado	Evidente	Alta
RFUN 1.3	Registrar o valor recebido para venda realizada	Evidente	Média
RFUN 1.4	Calcular e exibir o troco para o pagamento em dinheiro	Evidente	Alta

EOR	Versão:
Nome do Software	Data/Hora:

RFUN 1.5	Emitir recibo de pagamento de venda realizada	Evidente	Alta
RFUN 1.6	Atualizar o estoque do produto vendido	Oculto	Alta
RFUN 1.7	Registrar dados da venda realizada (data, hora, PDV, produtos vendidos...)	Oculto	Alta

Tabela 3 – Requisitos Funcionais do Caso de Uso CSU1-Processar Venda

4. 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.

4.1. Requisitos 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 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, vendas realizadas, comissões.	CSUx
RINF2	Cadastral	Produto: código, descrição, unidade de armazenamento, preço.	CSUy, CSUz
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

Tabela 4 – Requisitos de Informação

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

EOR	Versão:
Nome do Software	Data/Hora:

4.2 Requisitos/Restrições de Interface Homem-Computador (RIHC)

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.

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.	CSU1

Tabela 5 – Requisitos e Restrições de Interface Homem-Computador

4.3 Requisitos e Restrições 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.

EOR	Versão:
Nome do Software	Data/Hora:

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

Tabela 6 – Requisitos e Restrições de Interface Externa

4.4 Requisitos/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 (10/12/15), para testes em ambiente real.	CSUi, CSUj, CSUk

Tabela 7 – Requisitos e Restrições do Projeto

4.5 Requisitos/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	Devido a possibilidade de mudança de plataforma do sistema de rede local para internet, 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

Tabela 8 – Requisitos e Restrições do Arquitetura de Software

4.6 Requisitos/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) disponível na empresa	Todos

Tabela 9 – Requisitos e Restrições de Plataforma de Hardware

EOR	Versão:
Nome do Software	Data/Hora:

4.7 Requisitos/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

Tabela 10 – Requisitos e Restrições de Plataforma de Software

4.8 Requisitos/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

Tabela 11 – Requisitos e Restrições de Desempenho

4.9 Requisitos/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

EOR	Versão:
Nome do Software	Data/Hora:

não se restringem a eles. 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 e sete dias por semana	Todos

Tabela 12 – Requisitos e Restrições de Disponibilidade

4.10 Requisitos/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.

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

Tabela 13 – Requisitos e Restrições de Segurança

4.11 Requisitos/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);

EOR	Versão:
Nome do Software	Data/Hora:

- 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

Tabela 14 – Requisitos e Restrições de Manutenibilidade

4.12 Requisitos/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.

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

Tabela 15 – Requisitos e Restrições do Portabilidade

4.13 Requisitos/Restrições 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

Tabela 16 – Requisitos e Restrições de Documentação

EOR	Versão:
Nome do Software	Data/Hora:

5. Requisitos Futuros (RFUT)

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

Exemplo:

Ref.	Descrição	Casos de Uso
RFUT1	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

Tabela 17 – Requisitos Futuros

6. 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:

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

Tabela 18 – Requisitos Funcionais e Requisitos Não Funcionais Vinculados

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>

Tabela 19 – Requisitos Funcionais e suas Origens

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>

Tabela 20 – Requisitos Não Funcionais e suas Origens

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>

Tabela 21 – Casos de Uso e suas Origens

EOR	Versão:
Nome do Software	Data/Hora:

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.

7. Aprovação Formal

O documento 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.

8. 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 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 2.0, outubro de 2015.*

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

9. 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.