

**COLÉGIO ESTADUAL PROFESSORA RENI CORREIA GAMPER-EMPN**

**ANA PAULA HEINZEN DE AGUIAR**

**ANALISE DE REQUISITOS PARA UM  
CONSULTÓRIO ODONTOLÓGICO**

**MANOEL RIBAS/PR  
2012**

ANA PAULA HEINZEN DE AGUIAR

**ANALISE DE REQUISITOS PARA UM  
CONSULTÓRIO ODONTOLÓGICO**

Trabalho de conclusão de curso apresentado à disciplina de análise de sistema como requisito parcial para obtenção de nota do curso Técnico em Informática do Colégio Estadual Professora Reni Correia Gamper- EMPN.

Professora Orientadora: Rosana Diman Agonilha.

**MANOEL RIBAS/PR  
2012**

**FOLHA DE APROVAÇÃO**

ANA PAULA HEINZEN DE AGUIAR

**ANÁLISE DE REQUISITOS PARA UM  
CONSULTÓRIO ODONTOLÓGICO**

Trabalho de Conclusão de curso apresentado a disciplina como requisito parcial para obtenção de título de técnico em informática do Colégio Estadual Professora Reni Correia Gamper-EMPN.

Aprovado em \_\_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

---

---

---

**MANOEL RIBAS/PR  
2012**

## DEDICATÓRIA

Dedico este trabalho a minha família e  
amigos

## **AGRADECIMENTO**

A Deus por me dar força para seguir sempre em frente, não deixar me abater e caminhar ao meu lado.

A minha família por me apoiar nas minhas decisões e estarem sempre comigo.

Aos professores por me darem o conhecimento necessário não só referente ao curso, mas a vida também.

Ao professor orientador por me estimular e ajudar nesta fase da minha vida.

Aos meus amigos por estarem sempre ao meu lado nos momentos em que mais preciso.

Fazei-me justiça, Senhor, pois tenho andado retamente e, confiando em vós, não vacilei. Sondai-me, senhor, e provai-me; escutai meus rins e meu coração. Tenho sempre diante dos olhos vossa bondade e caminho na vossa verdade (Salmo-25, 1-3).

## LISTA DE FIGURAS

Figura 01- Representação de um tipo de objeto.....	20
Figura 02- Representação de Relacionamento.....	20
Figura 03- Representação de um Diagrama de Entidade e Relacionamento.....	21
Figura 04- Representação de um processo.....	22
Figura 05- Representação de um fluxo.....	23
Figura 06- Representação de um depósito.....	23
Figura 07- Representação de um terminador.....	23
Figura 08- Representação do processo “Cadastrar”.....	24
Figura 09- Representação do processo “Pedido/Fornecedor”.....	24
Figura 10- Representação do processo “Agendar”.....	24
Figura 11- Representação do processo “Atendimento”.....	24
Figura 12- Representação do processo “Caixa”.....	25
Figura 13- Representação de um Diagrama Geral do Sistema.....	28
Figura 14- Representação de um ator.....	29
Figura 15- Representação de uma interação em caso de uso.....	30
Figura 16- Representação de um diagrama de caso de uso.....	30
Figura 17- Representação de um protótipo.....	31

## SUMARIO

<b>1 INTRODUÇÃO</b> .....	<b>9</b>
<b>2 JUSTIFICATIVA</b> .....	<b>10</b>
<b>3 OBJETIVOS</b> .....	<b>11</b>
3.1 OBJETIVO GERAL.....	11
3.2 OBJETIVOS ESPECÍFICOS .....	11
<b>4 CICLO DE VIDA DE UM SISTEMA</b> .....	<b>12</b>
4.1 FASES DE DEFINIÇÃO .....	12
<b>4.1.1 Análise e Especificação</b> .....	<b>12</b>
<b>4.1.2 Estudo de Viabilidade</b> .....	<b>13</b>
4.2 FASE DE DESENVOLVIMENTO .....	14
<b>4.2.1 Design</b> .....	<b>14</b>
<b>4.2.2 Implementação e Integração</b> .....	<b>14</b>
<b>4.2.3 Verificação e Validação</b> .....	<b>15</b>
4.3 FASES DE OPERAÇÃO .....	15
<b>4.3.1 Distribuição, Instalação e Configuração</b> .....	<b>15</b>
<b>4.3.2 Utilização e administração</b> .....	<b>16</b>
<b>4.3.3 Manutenção- Corretiva, Evolutiva e Adaptativa</b> .....	<b>16</b>
4.4 FASES DE RETIRADA.....	16
<b>4.4.1 Migração, reengenharia, engenharia reversa</b> .....	<b>16</b>
<b>5 ANÁLISE DE REQUISITOS</b> .....	<b>17</b>
<b>6 ENTREVISTAS ESTRUTURADAS</b> .....	<b>17</b>
<b>7 QUESTIONÁRIOS</b> .....	<b>19</b>
<b>8 DIAGRAMA DE ENTIDADES E RELACIONAMENTOS (DER)</b> .....	<b>20</b>
7.1 COMPONENTES DE UM DER .....	20
<b>8.1.1 Tipos de objetos (Entidades)</b> .....	<b>20</b>
<b>8.1.2 Relacionamentos</b> .....	<b>20</b>
<b>9 DIAGRAMA DE FLUXO DE DADOS</b> .....	<b>22</b>
9.1 COMPONENTES DE UM DFD .....	22
<b>9.1.1 Processo</b> .....	<b>22</b>
<b>9.1.2 Fluxo</b> .....	<b>23</b>
<b>9.1.3 Deposito</b> .....	<b>23</b>
<b>9.1.4 Entidades</b> .....	<b>23</b>
<b>10 DICIONÁRIO DE DADOS</b> .....	<b>26</b>
<b>11 DIAGRAMA GERAL DO SISTEMA</b> .....	<b>28</b>
<b>12 DIAGRAMA DE CASO DE USO</b> .....	<b>29</b>
12.1 COMPONENTES DE UM DIAGRAMA DE CASO DE USO .....	29
<b>12.1.1 Ator</b> .....	<b>29</b>
<b>12.1.2 Interação em caso de uso</b> .....	<b>30</b>
<b>13 PROTOTIPAGEM DE TELA</b> .....	<b>31</b>
<b>CONCLUSÃO</b> .....	<b>32</b>
<b>REFERÊNCIA BIBLIOGRÁFICA</b> .....	<b>33</b>



## 1 INTRODUÇÃO

A tecnologia está evoluindo cada vez mais, isso se vê desde um simples ato de cozinhar, até uma sofisticada operação cirúrgica, e logicamente que com um consultório odontológico não é diferente, tanto na parte de equipamentos quanto na de sistemas a informatização é necessária e quase que obrigatória.

A análise de requisitos é muito importante para criação de um sistema, é com ela que se obtêm as informações que irão conter no software, através do levantamento das necessidades que este irá requerer.

O trabalho trata primeiramente do tema Ciclo de vida de sistema que por fases mostra desde a concepção até a morte do software. As fases são: Definição; Desenvolvimento; Operação e Retirada. Após isso é descrito os diagramas e os meios de levantamento de necessidades que foram utilizados na análise de requisitos que são: Entrevista e Questionário; Diagrama de Entidades e Relacionamentos (DER); Diagrama de Fluxo de Dados (DFD); Dicionário de Dados; Diagrama Geral do Sistema; Diagrama de Caso de Uso e Prototipagens de Telas.

Com isso o trabalho aqui apresentado irá demonstrar o processo de criação de um sistema, o qual possui como intuito o controle de todas as atividades envolvidas em um consultório odontológico.

## 2 JUSTIFICATIVA

A análise de requisitos através de suas pesquisas e argumentos tem como base os dados necessários para a elaboração de um sistema, é com as informações adquiridas que o programador pode construir um programa que seja completo e preciso nas suas necessidades.

A fase de análise de requisitos é a mais complexa, porém, uma das fases mais importante, pois, sem ela o programa não sairia nem do papel. Com base nos questionários e na entrevista é que o analista localiza todos os requisitos necessários para a elaboração das etapas que compõem um sistema, para que o software seja coerente e eficaz.

É uma profissão muito aceita no mercado, pois a tecnologia esta atingindo cada vez mais todos os estabelecimentos comerciais, assim aderindo a ela informatizando suas atividades para se tornarem cada vez mais fáceis e econômicos seus atendimentos. Portanto se torna bem mais simples encontrar emprego nessa área tecnológica em pequenas cidades ou em grandes centros.

Escolhi esta disciplina, pois é uma das que mais me identifico, e pretendo me especializar ainda mais nela e assim me tornar uma analista de sistema.

### 3 OBJETIVOS

#### 3.1 OBJETIVO GERAL

Realizar uma análise de requisitos de um consultório odontológico, tendo como ponto de partida o levantamento de dados, e a elaboração de diagramas para a melhor visualização dos dados internos do sistema, posteriormente a elaboração de protótipos de telas possíveis, com o desenvolvimento de um sistema.

#### 3.2 OBJETIVOS ESPECÍFICOS

- Elaborar entrevistas e questionários como fonte de levantamento de dados;
- Analisar os questionários e os resultados das entrevistas;
- Entender quais as necessidades básicas propostas para a aplicação de um sistema.
- Elaborar diagrama de entidades e relacionamentos; diagrama de fluxo de dados; dicionário de dados; diagrama geral do sistema; diagrama de caso de uso e os protótipos das principais telas.

## **4 CICLO DE VIDA DE UM SISTEMA**

Segundo Yourdon (1990), o ciclo de vida de um projeto de sistema é o modo como o projeto é desenvolvido na empresa e é uma maneira simples para que qualquer pessoa da área de desenvolvimento de sistemas possa entrosar-se com o projeto a ser desenvolvido.

([http://www.abepro.org.br/biblioteca/ENEGEP2001\\_TR93\\_0290.pdf](http://www.abepro.org.br/biblioteca/ENEGEP2001_TR93_0290.pdf)).

O ciclo de vida contém varias concepções, a qual será descrita a seguir é dividida em quatro fases: Definição; Desenvolvimento; Operação e Retirada.

### **4.1 FASES DE DEFINIÇÃO**

A fase de definição do software ocorre em conjunto com outras atividades como a modelagem de processos de negócios e análise de sistemas. Nesta atividade, diversos profissionais buscam o conhecimento da situação atual e a identificação de problemas para que possam elaborar propostas de solução de sistemas computacionais que resolvam tais problemas. Dentre as propostas apresentadas, deve-se fazer um estudo de viabilidade, incluindo análise custo-benefício, para se decidir qual solução será a escolhida.

<http://engenhariadesoftware.blogspot.com.br/2007/02/ciclo-de-vida-do-software-parte-1.html>

#### **4.1.1 Análise e Especificação**

A análise é o processo de observação e levantamento dos elementos do ambiente onde o software será implantado. Devem-se identificar as pessoas que terão contato com o software, quer seja um contato operacional ou para o fornecimento de informações relevantes para o seu desenvolvimento.

A especificação é a descrição sistemática e abstrata do que o software deve fazer a partir daquilo que foi analisado anteriormente. Ela apresenta a solução de como os problemas levantados na análise devem ser resolvidos pelo software em

desenvolvimento. <http://www.pmkb.com.br/artigos-mainmenu-25/3810-analise-e-especificacoes-de-requisitos.html>

#### **4.1.2 Estudo de Viabilidade**

Uma vez que o processo de desenvolver um importante sistema de informação pode ser dispendioso, a etapa de investigação de sistemas frequentemente exige um estudo preliminar chamado de estudo de viabilidade. O estudo de viabilidade é um estudo preliminar que investiga as necessidades de informação dos usuários esperados e determina os requisitos de recursos, custos, benefícios e viabilidade de um projeto proposto. <http://rogeroli.sites.uol.com.br/sistinform.htm>

Devido ao conhecimento altamente especializado que esta etapa requer, lança-se mão de especialistas financeiros para se desenvolver um estudo claro de viabilidade do novo sistema em termos econômicos, e dos benefícios que ele traria sob o ponto de vista financeiro.

A viabilidade de um sistema pode ser avaliada em termos de quatro categorias principais:

- Viabilidade Organizacional: concentra-se no grau de eficácia com que um sistema de informação proposto apoia os objetivos da organização e seu plano estratégico para os sistemas de informação.
- Viabilidade Econômica: concentra-se em saber se os custos e benefícios tangíveis do sistema proposto excederão os custos de desenvolvimento e operação do sistema.
- Viabilidade Técnica: concentra-se na confiança/capacidade do hardware e software de satisfazer as necessidades do sistema proposto e se eles podem ser adquiridos ou desenvolvidos no tempo requisitado.
- Viabilidade Operacional: concentra-se na disposição e capacidade da administração, funcionários, clientes, fornecedores e outros de operar, utilizar e apoiar o sistema proposto.

<http://rogeroli.sites.uol.com.br/sistinform.htm>

## 4.2 FASE DE DESENVOLVIMENTO

A fase de desenvolvimento ou de produção do software abrange todas as atividades que tem por objetivo a construção do produto. Ela inclui principalmente o design, a implementação e a verificação e validação do software.

### 4.2.1 Design

A atividade de design compreende todo o esforço de concepção e modelagem que têm por objetivo descrever como o software será implementado.

- Design conceitual: envolve a elaboração das idéias e conceitos básicos que determinam os elementos fundamentais do software em questão.
- Design da interface de usuário: envolve a elaboração da maneira como o usuário pode interagir para realizar suas tarefas, a escolha dos objetos de interfaces (botões, menus, caixas de texto, etc.),
- Design de arquitetura de software: deve elaborar uma visão macroscópica do software em termos de componentes que interagem entre si.
- Design de algoritmos e estrutura de dados, também conhecido como design detalhado, visa determinar, de maneira independente da linguagem de programação adotada, as soluções algorítmicas e as estruturas de dados associados.

<http://engenhariadesoftware.blogspot.com.br/2007/02/ciclo-de-vida-do-software-parte-1.html>

### 4.2.2 Implementação e Integração

A implementação envolve as atividades de codificação, compilação, integração e testes. A codificação visa traduzir o design num programa, utilizando linguagens e ferramentas adequadas. A codificação deve refletir a estrutura e o comportamento descrito no design. Os componentes arquiteturais devem ser

codificados de forma independente e depois integrados. Os testes podem ser iniciados durante a fase de implementação. A depuração de erros ocorre durante a programação utilizando algumas técnicas e ferramentas. É fundamental um controle e gerenciamento de versões para que se tenha um controle correto de tudo o que está sendo codificado. <http://engenhariadesoftware.blogspot.com.br/2007/02/ciclo-de-vida-do-software-parte-1.html>

### **4.2.3 Verificação e Validação**

Verificação e validação destinam-se a mostrar que o sistema está de acordo com a especificação e que ele atende às expectativas de clientes e usuários. A validação visa assegurar se o programa está fazendo aquilo que foi definido na sua especificação. A verificação visa verificar se o programa está correto, isto é, não possui erros de execução. Existem diferentes formas de verificação e validação. Inspeção analítica e revisão de modelos, documentos e código fonte são formas que podem ser usadas antes mesmo que o programa seja completamente codificado. Os testes de correção, desempenho, confiabilidade, robustez, usabilidade, dentre outros, visam avaliar diversos fatores de qualidade a partir da execução do software. Diferentes técnicas de testes podem ser aplicadas para cada um destes fatores. <http://engenhariadesoftware.blogspot.com.br/2007/02/ciclo-de-vida-do-software-parte-1.html>

## **4.3 FASES DE OPERAÇÃO**

A fase de operação envolve diferentes tipos de atividades:

### **4.3.1 Distribuição, Instalação e Configuração**

A distribuição e entrega pode ser feita diretamente pelo desenvolvedor (em caso de software personalizado), ou em um pacote a ser vendido em prateleiras de lojas ou para ser baixado pela Internet (em caso de softwares genéricos). O processo de instalação e configuração, normalmente, pode ser feito com a ajuda de

software de instalação disponibilizados pelos fabricantes dos ambientes operacionais.

### **4.3.2 Utilização e administração**

A atividade de utilização é o objeto do desenvolvimento do software. A qualidade da utilização é a usabilidade do software.

### **4.3.3 Manutenção- Corretiva e Evolutiva**

A manutenção normalmente ocorre de duas formas: corretiva e evolutiva. A manutenção corretiva visa à resolução de problemas referentes à qualidade do software (falhas, baixo desempenho, baixa usabilidade, falta de confiabilidade, etc.). A manutenção evolutiva ou adaptativa visa à produção de novas versões do software de forma a atender a novos requisitos dos clientes, ou adaptar-se às novas tecnologias que surgem (hardware, plataformas operacionais, linguagens, etc). Mudanças no domínio de aplicação implicam em novos requisitos e incorporação de novas funcionalidades. Surgimento de novas tecnologias de software e hardware e mudanças para uma plataforma mais avançada também requerem evolução.

<http://engenhariadesoftware.blogspot.com.br/2007/02/ciclo-de-vida-do-software-parte-1.html>

## **4.4 FASES DE RETIRADA**

A fase retirada é um grande desafio para os tempos atuais. Diversos softwares que estão em funcionamento em empresas possuem excelentes níveis de confiabilidade e de correção. No entanto, eles precisam evoluir para novas plataformas operacionais ou para a incorporação de novos requisitos.

### **4.4.1 Migração, reengenharia, engenharia reversa**

Processos de reengenharia podem ser aplicados para viabilizar a transição ou migração de um software legado para um novo software de forma a proporcionar uma retirada mais suave.



## 5 ANÁLISE DE REQUISITOS

A análise de requisitos é a documentação do software, ela contribui para que o programador não se perca ou deixe alguma informação necessária passar por despercebido. Ela constitui varias etapas que o analista fará segundo seu critério. Os atributos (são dados das entidades ou relacionamentos) são encontrados graças a ela.

Também chamada de Análise de Sistemas ou Engenharia de Requisitos. Parte dos requisitos obtidos e revisados e do Plano de Projeto. Visa fornecer representações (modelos) de "Informação" e de "função" que:

- Possam ser transformadas (traduzidas) em "projeto" de dados, arquitetura, interface e procedimentos;
- Sirvam como um meio de determinar a qualidade do software.  
<http://www.paiossin.com/wordpress/wp-content/uploads/2011/11/Anlise-de-Requisitos-Conceitos.pdf>

Depois da análise completamente pronta, ela é entregue ao programador para que o mesmo configure o sistema.

## 6 ENTREVISTAS ESTRUTURADAS.

A entrevista é a primeira fase da análise, é feita diretamente com o proprietário da empresa envolvida ou algum funcionário da tal, juntamente com o analista. As perguntas são elaboradas para a definição dos atributos, é com elas que o analista começa a levantar os dados necessários para o software. Deverá especificar detalhadamente a vida da empresa com todos os seus detalhes para que todas as atividades sejam altamente contadas no sistema.

A seguir teremos alguns passos para realizar a entrevista:  
([http://sim2008.files.wordpress.com/2009/09/as\\_tag\\_23-11-2009.pdf](http://sim2008.files.wordpress.com/2009/09/as_tag_23-11-2009.pdf))

- Selecionar os entrevistados.
- Desenvolver questões da entrevista.
- Estabelecer objetivos da entrevista.
- Conduzir a entrevista.

## 7 QUESTIONÁRIOS

O questionário é uma parte essencial na análise de requisitos, é com ela que o analista encontra as informações necessárias para os demais passos. As perguntas são elaboradas conforme a empresa que requereu tais serviços.

O questionário é deixado na empresa e será respondido por um funcionário ou o próprio proprietário. Depois de alguns dias o analista volta à empresa para a devolução do questionário já com as respostas necessárias. E então as respostas serão analisadas e as informações certas serão usadas no levantamento de requisitos.

Um questionário pode ser entendido mediante as definições a seguir expostas ([http://sim2008.files.wordpress.com/2009/09/as\\_tag\\_23-11-2009.pdf](http://sim2008.files.wordpress.com/2009/09/as_tag_23-11-2009.pdf)):

- Conjunto de questões escritas, usualmente enviadas para um grande número de pessoas.
- Podem ser em formato de papel ou eletrônico.
- Selecionar participantes representativos
- Desenvolver questões claras e de fácil análise.
- Definir estratégias para obter um bom número de respostas.

## 8 DIAGRAMA DE ENTIDADES E RELACIONAMENTOS (DER)

É modelo em rede que descreve a diagramação dos dados armazenados de um sistema em alto nível de abstração.

Para o analista de sistema o DER é um importante benefício: ele realça os relacionamentos entre os depósitos de dados de um DFD que de outro modo só seriam percebidos nas especificações de processos.

### 8.1 COMPONENTES DE UM DER

#### 8.1.1 Tipos de objetos (Entidades)

É representado por um retângulo. Ele representa uma coleção ou um conjunto de objetos (coisas) do mundo real cujos membros individuais.



Figura 01- Representação de um tipo de objeto

#### 8.1.2 Relacionamentos

Um relacionamento representa um conjunto de conexões entre objetos e é representado por um losango.



Figura 02- Representação de relacionamento

(YOURDON, 1990, p.)

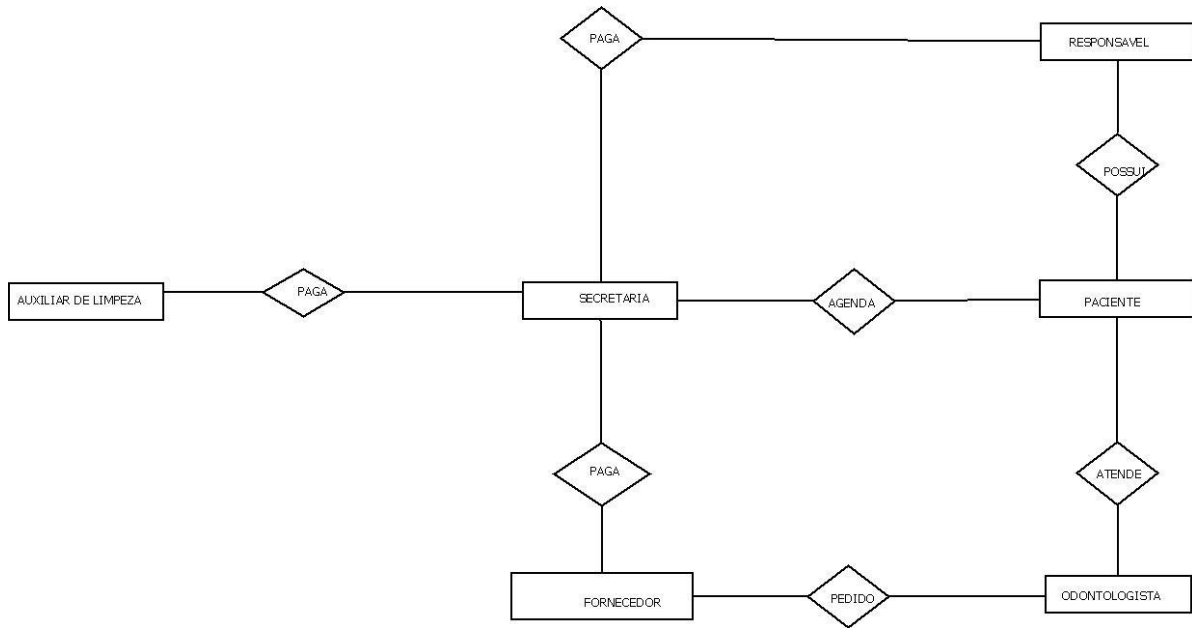


Figura 03-Representação do diagrama e entidade de relacionamento

## 9 DIAGRAMA DE FLUXO DE DADOS

Esse diagrama é uma ferramenta de modelagem que nos permite imaginar um sistema como uma rede de processos funcionais, interligados por “duto” e “tanques de armazenamentos” de dados.

### 9.1 COMPONENTES DE UM DFD

#### 9.1.1 Processo

O primeiro componente de um DFD é conhecido como processo. Os sinônimos mais conhecidos são bolha, função e transformação. O processo mostra uma parte do sistema, a que transforma entradas em saídas - isto é, mostra como uma ou mais entradas são convertidas em saídas. O processo é representado graficamente por um círculo. Alguns analistas de sistemas preferem usar um oval, ou um retângulo de vértices curvos, como mostrado na figura; outros preferem ainda um retângulo. As diferenças entre esses três formatos são puramente cosméticas, embora seja obviamente importante utilizar o mesmo formato de maneira consistente para representar todas as funções do sistema (YOURDON, 2005, p. 2).

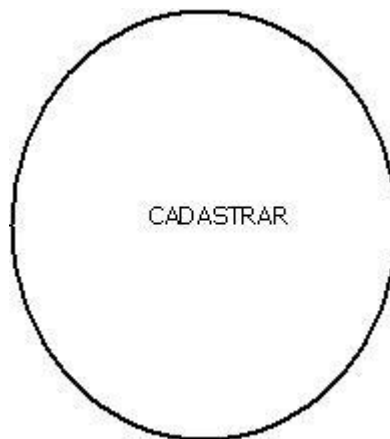


Figura 04-Representação de um processo

### 9.1.2 Fluxo

O fluxo é utilizado para mostrar o movimento de fragmentos ou de pacotes de informações de um ponto a outro do sistema. É representado graficamente por uma seta que entra ou sai de um processo.



Figura 05-Representação de um fluxo

### 9.1.3 Depósito

O depósito é utilizado para se modelar uma coleção de pacotes de dados em repouso.

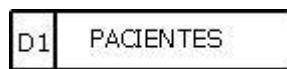


Figura 06-Representação de um depósito

### 9.1.4 Entidades

Os terminadores representam entidades externas com as quais o sistema se comunica.



Figura 07-Representação de um terminador

(YOURDON, 1990,)

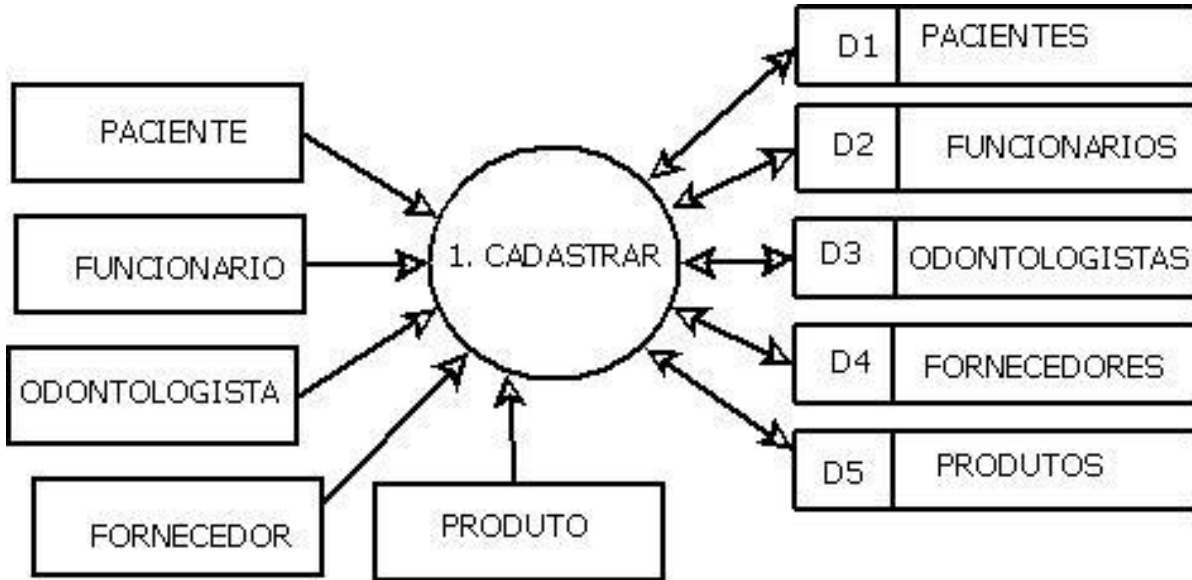


Figura 08-Representação do processo “cadastrar”

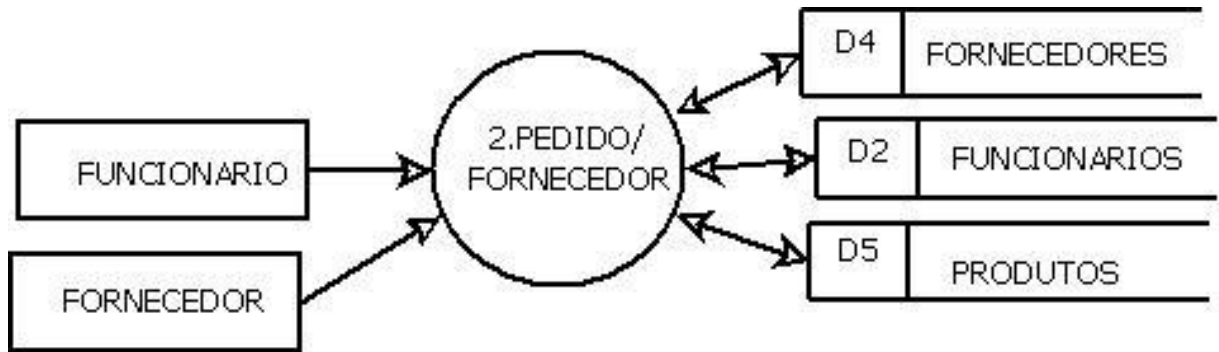


Figura 09-Representação do processo “Pedido/Fornecedor”

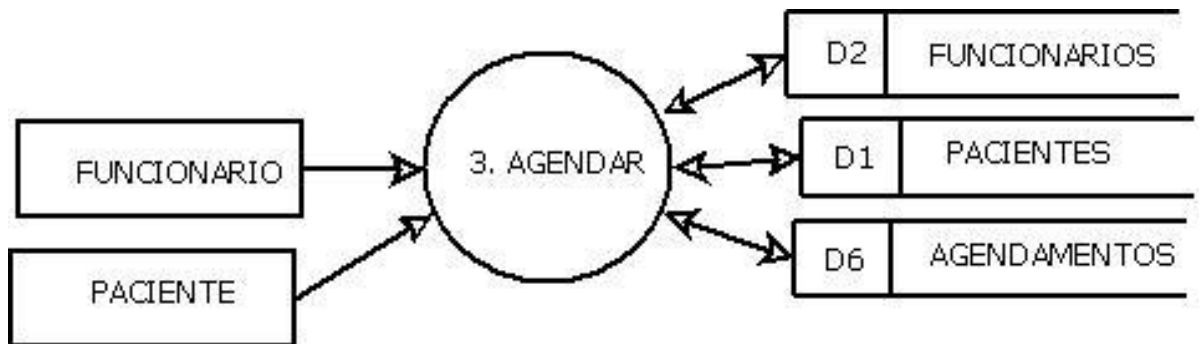


Figura 10-Representação do processo “agendar”



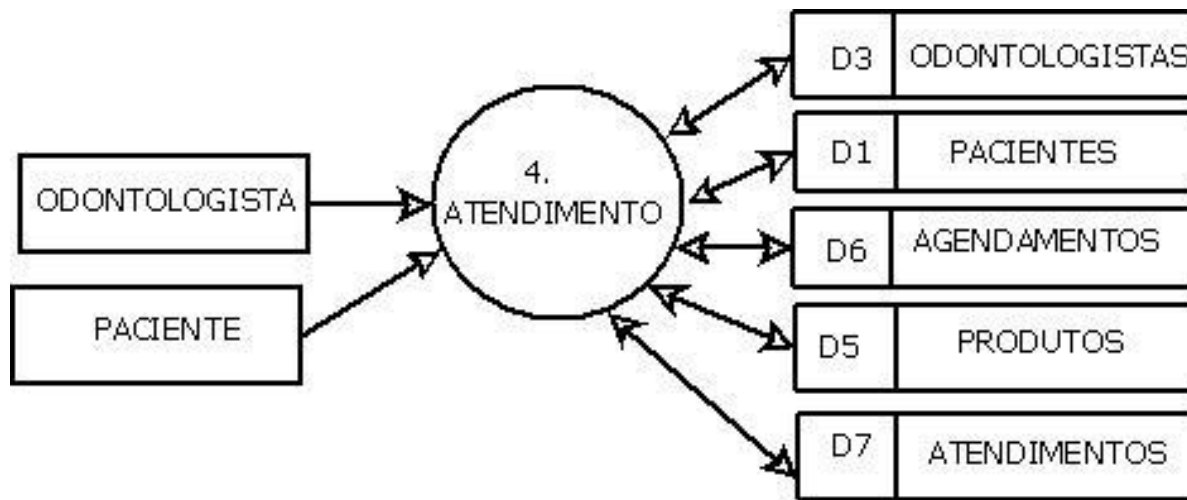


Figura 11-Representação do processo "Atendimento"

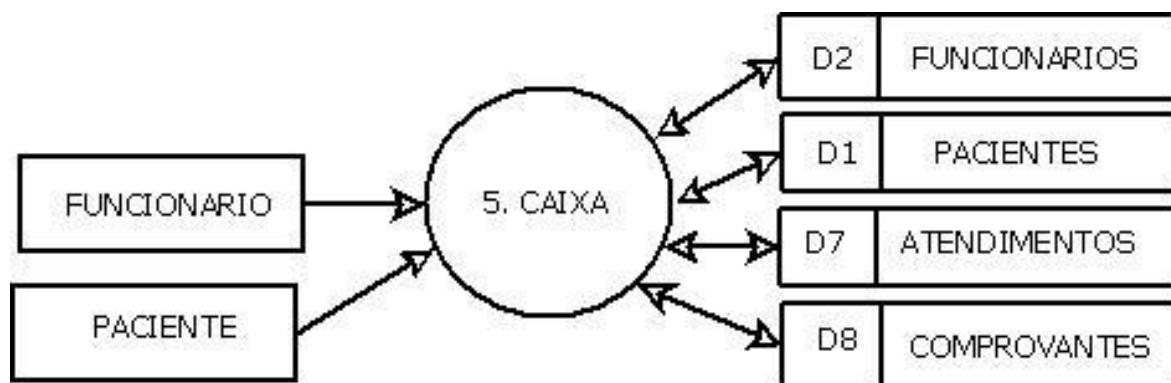


Figura 12-Representação do processo "Caixa"

## 10 DICIONÁRIO DE DADOS

O dicionário de dados é uma listagem organizada de todos os elementos de dados pertinentes ao sistema, com definições precisas e rigorosas para que o usuário e o analista de sistema possam conhecer todas as entradas, saídas, componentes de depósitos e cálculos intermediários. (YOURDON,1990)

- Número: Indica o numero do elemento;
- Nome Campo: Nome do atributo;
- Tipo: Indica o tipo do produto;
- Tamanho: Colocar o tamanho do campo quando o mesmo for do tipo caractere, inteiro ou real;
- Descrição: Informar, de forma clara, uma descrição textual explicando o atributo;
- PK: Identifica a chave primaria.

Cada entrada no DD é constituída por um identificador e respectiva descrição.

Segundo o site [http://www.estv.ipv.pt/paginaspeessoais/ajas/AS/Apontamentos%20Te%C3%B3ricos/as\\_3\\_4.pdf](http://www.estv.ipv.pt/paginaspeessoais/ajas/AS/Apontamentos%20Te%C3%B3ricos/as_3_4.pdf)) A descrição de cada entrada inclui:

- O seu significado;
- O seu conteúdo (só para dados compostos);
- Os valores permitidos e unidades (só para dados elementares);
- A chave primaria (só para depósitos de dados).

A seguir será apresentado um exemplo de dicionário de dados.

Nº	NOME CAMPO	TIPO	TAM	DESCRIÇÃO	PK
1	Código_pac	N	6	Código do paciente	*
2	Nome_pac	A	40	Nome completo do paciente	
3	RG_pac	A	15	Número do registro geral do paciente	
4	Bairro_pac	A	15	Bairro do paciente	

5	Endereço_pac	A	30	Endereço completo do paciente com numero	
6	CPF_pac	A	20	Número do cadastro da pessoa física do paciente	
7	Cidade_pac	A	20	Cidade do paciente	
8	UF_pac	A	2	Unidade Federativa do paciente	
9	Datacad_pac	Date	8	Data de cadastro do paciente	
10	Datanasc_pac	Date	8	Data de nascimento do paciente	
11	Idade_pac	N	3	Idade do paciente	
12	Responsável_pac	A	45	Responsável pelo paciente	
13	Telefone_pac	A	13	Número do telefone completo com DDD do paciente	
14	Celular_pac	A	13	Número do celular completo com DDD do paciente	
15	Sexo_pac	A	1	Sexo do paciente	

## 11 DIAGRAMA GERAL DO SISTEMA

Também conhecido por Diagrama de Módulos, apresenta os módulos do sistema, as ligações entre eles, os seus submódulos e/ou itens. A representação do sistema de um consultório odontológico está representada abaixo:

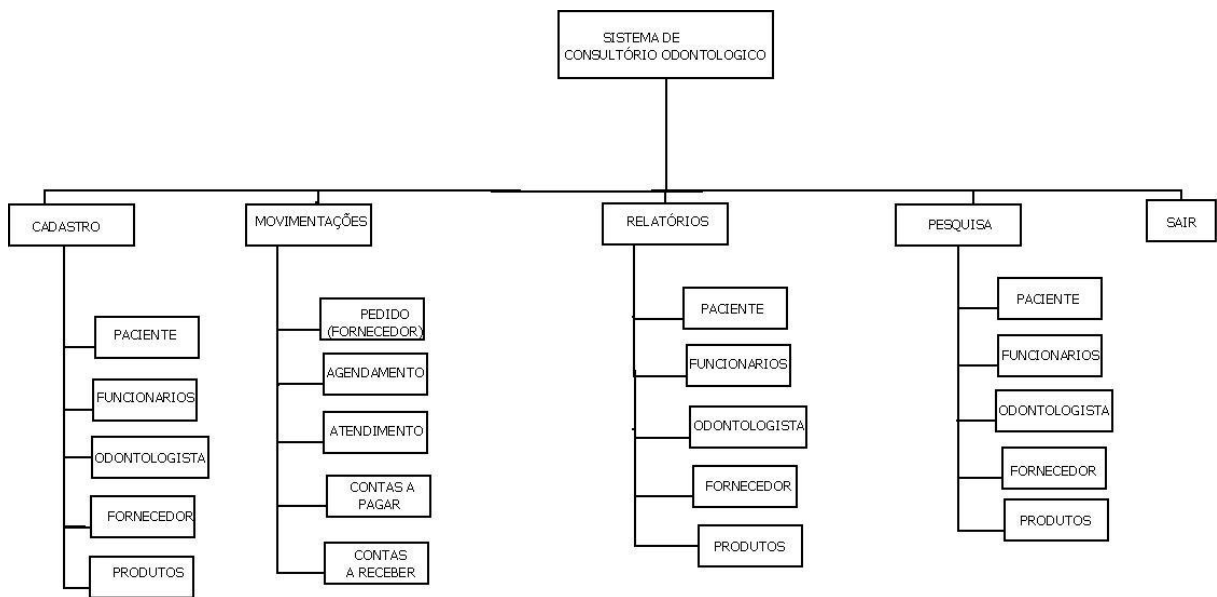


Figura 13-Representação de diagrama geral do sistema

## 12 DIAGRAMA DE CASO DE USO

A modelagem de um diagrama de caso de uso é uma técnica usada para descrever e definir os requisitos funcionais de um sistema. Eles são escritos em termos de atores externos, caso de uso e o sistema modelado. Os atores representam o papel de uma entidade externa ao sistema como um usuário, um hardware, ou outro sistema que interage com o sistema modelado. Os atores iniciam a comunicação com o sistema através dos use-cases, onde o caso de uso representa uma sequência de ações executadas pelo sistema e recebe do ator que lhe utiliza dados tangíveis de um tipo ou formato já conhecido, e o valor de resposta da execução de um use-case (conteúdo) também já é de um tipo conhecido, tudo isso é definido juntamente com o use-case através de texto de documentação.

### 12.1 COMPONENTES DE UM DIAGRAMA DE CASO DE USO

#### 12.1.1 Ator

Especifica um papel executado que interage com o cenário (caso de uso). Um ator deve ter associações exclusivamente para casos de uso. A exceção é um ator que possa herdar o papel de outro. Um ator é representado por um boneco (stick man).

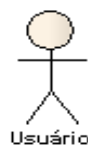


Figura 14-Representação de um ator

Um ator pode ser um usuário, um humano, uma máquina, hardware, uma aplicação. Um ator deve representar uma interação com o sistema. Para identificar um ator de um sistema podemos fazer as seguintes perguntas (<http://techblog.desenvolvedores.net/2011/05/12/diagrama-de-caso-de-uso-uml/>):

- Quem está interessado na exigência?
- O sistema usa um recurso externo?

- Quem fornecer a informação irá usar e modificar, ou às removerá?
- Uma pessoa representa um papel?
- O sistema interage com um sistema legado?

### 12.1.2 Interação em caso de uso

O ator comunica-se com o sistema através do envio e recebimento de mensagens. Um ator comunica-se com o caso de uso, esta comunicação é mostrada conectando-se o símbolo do ator ao símbolo do caso de uso por um caminho sólido. (<http://techblog.desenvolvedores.net/2011/05/12/diagrama-de-caso-de-uso-uml/>).



Figura 15-Representação de uma interação em caso de uso

Veja o modelo a seguir:

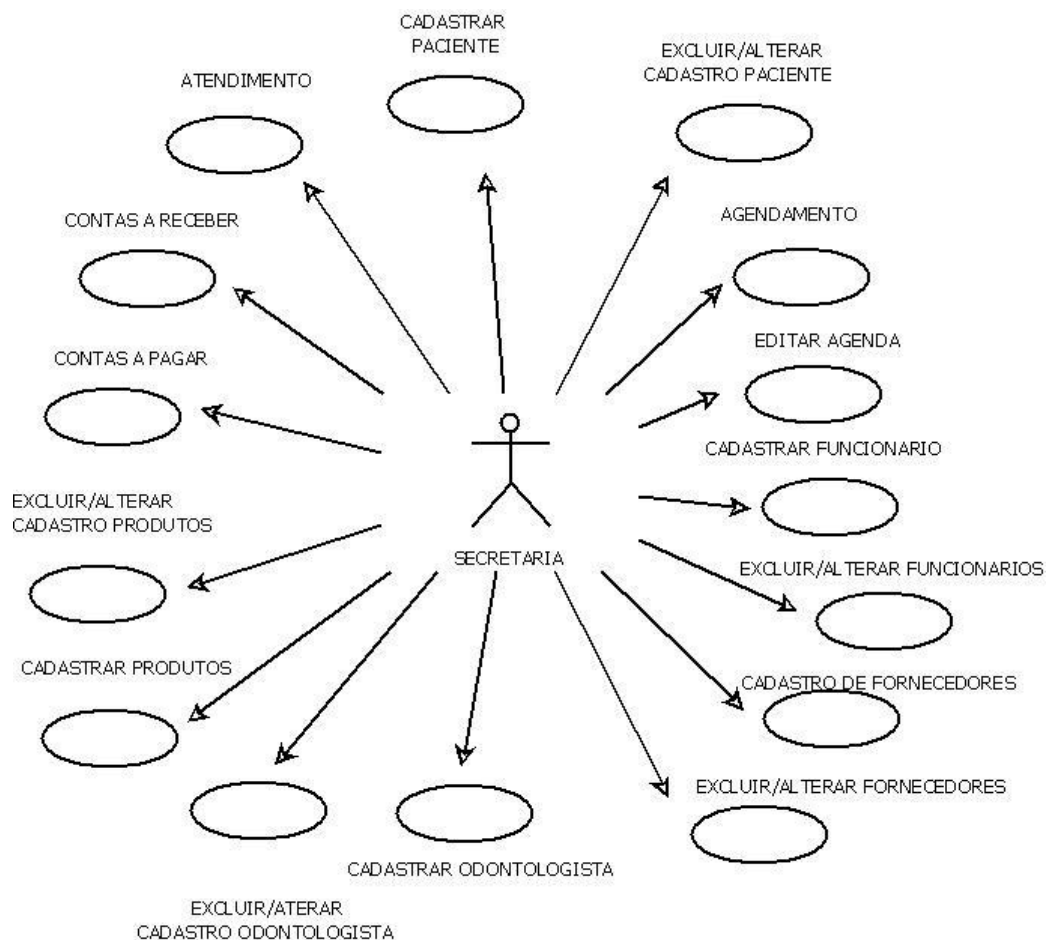


Figura 16-Representação de diagrama de caso de uso

### 13 PROTOTIPAGEM DE TELA

É uma técnica que serve de complemento à Análise de Requisitos. Um protótipo é um esboço de alguma parte do sistema. Podem ser construídas para telas de entrada, telas de saída, subsistemas e até mesmo o sistema como um todo. Utiliza as linguagens de programação visual.

Um protótipo é construído após o levantamento de requisitos e serve para serem validados. É visto por um ou mais usuários que fazem críticas acerca de uma ou outra característica e pode ser corrigido ou refinado de acordo com essas críticas. O processo de revisão e refinamento continua até o protótipo ser aceito pelos usuários. [http://rubiadenardi.pro.br/apostila\\_analise\\_sistemas.pdf](http://rubiadenardi.pro.br/apostila_analise_sistemas.pdf)

A característica principal desse modelo é gerar protótipos do sistema com definições de requisitos dadas pelo cliente. Essas definições geram documentos que, por sua vez resultam no protótipo. Esse protótipo é então testado pelo cliente para validar suas funcionalidades. <http://pt.scribd.com/doc/34613739/11/Modelo-de-Prototipagem>

**Cadastro de Paciente**

INERIR SALVAR CANCELAR ALTERAR EXCLUIR

◀ ◁ ▷ ▶

**Código** **DATA CADASTRO**

**NOME**

**DATA NASC**  **RG**  **CPF**

**ENDEREÇO**

**BAIRRO**  **CIDADE**  **UF**

**IDADE**  **SEXO**  
 F  M

**RESPONSÁVEL**

**TELEFONE**  **CELULAR**

Figura 17-Representação de um prototipo de tela

## **CONCLUSÃO**

Ao término deste trabalho ficou claro que a análise de requisitos é extremamente necessária para busca de dados corretos na criação de um software de alta qualidade e eficiência. Um sistema depende das informações que o futuro proprietário e usuário possuem, para que seja de total coerência nas suas aplicações.

A análise de requisitos depois do levantamento dos dados necessários e documentados de forma correta é importante para que o programador possa interpretá-los e construir o software. Hoje em dia tudo precisa de documentação, para registros e melhor clareza de ideias, desta forma também o odontologista precisará de uma ficha para suas consultas.

Portanto a análise de requisitos é a parte central de um sistema, o software só irá ser perfeito quando o levantamento dos dados for suficientemente bom para dar o respaldo necessário, atendendo de forma eficiente as necessidades de controle do fluxo de dados e geração de informações, não somente de um consultório odontológico, mas de toda empresa.



## REFERÊNCIA BIBLIOGRÁFICA

ALVES, Rafael Ferreira. **Ciclo de Vida de Desenvolvimento de Sistemas - Visão Conceitual dos Modelos Clássico, Espiral e Prototipação**. Disponível Em [Http://Www.Abepro.Org.Br/Biblioteca/Enegep2001\\_Tr93\\_0290.Pdf](Http://Www.Abepro.Org.Br/Biblioteca/Enegep2001_Tr93_0290.Pdf) . Acessado Em 25 Mai 2012.

**Análise de Sistema**. Disponível em [http://sim2008.files.wordpress.com/2009/09/as\\_tag\\_23-11-2009.pdf](http://sim2008.files.wordpress.com/2009/09/as_tag_23-11-2009.pdf) . Acessado em 18 mai 2012.

Curso técnico em informática Monteiro Lobato. **Introdução a Análise de Sistemas**.<http://pt.scribd.com/doc/34613739/11/Modelo-de-Prototipagem>. Acessado em 18 mai 2012.

DA SILVA, João Guilherme Pereira. **Análise e Especificações de Requisitos**. Disponível em <http://www.pmkb.com.br/artigos-mainmenu-25/3810-analise-e-especificacoes-de-requisitos.html>. Acessado em 11 mai 2012.

**Diagrama de Caso de Uso**. Disponível em <http://techblog.desenvolvedores.net/2011/05/12/diagrama-de-caso-de-uso-uml/>. Acessado em 23 mai 2012.

**Dicionário de dados (DD)**. Disponível em [http://www.estv.ipv.pt/paginaspessoais/ajas/AS/Apontamentos%20Te%C3%B3ricos/as\\_3\\_4.pdf](http://www.estv.ipv.pt/paginaspessoais/ajas/AS/Apontamentos%20Te%C3%B3ricos/as_3_4.pdf). Acessado em 18 mai 2012.

**Desenvolvendo Soluções com Sistemas de Informação**. Disponível em <http://rogeroli.sites.uol.com.br/sistinform.htm>. Acessado em 11 mai 2012.

LEITE, Jair C. **Ciclo de Vida do Software**. Disponível em <http://engenhariadesoftware.blogspot.com.br/2007/02/ciclo-de-vida-do-software-parte-1.html> . Acessada em 27 abr 2012.

PORTELLA, Cristiano R R. **Análise de requisitos conceito**. Disponível em <http://www.paiossin.com/wordpress/wp-content/uploads/2011/11/Anlise-de-Requisitos-Conceitos.pdf>. Acessada em 30 mai 2012.

SOUZA NETO, Oscar Nogueira de **Análise Comparativa das Metodologias de Desenvolvimento de Softwares Tradicionais e Ágeis**. Disponível em <http://pt.scribd.com/doc/34613739/11/Modelo-de-Prototipagem>. Acessado em 18 mai 2012.

YOURDON, Edward. **Análise estruturada moderna**. Rio de Janeiro: Campus, 1990.