

UNIVERSIDADE DO SAGRADO CORAÇÃO

MURILO ROSA DE MATTOS

**GESTÃO DE CONHECIMENTO: DESENVOLVIMENTO
DE UM SISTEMA BASEADO EM ONTOLOGIA**

BAURU

2016

MURILO ROSA DE MATTOS

**GESTÃO DE CONHECIMENTO: DESENVOLVIMENTO
DE UM SISTEMA BASEADO EM ONTOLOGIA**

Trabalho de Conclusão de Curso apresentado ao centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de bacharel em Ciência da Computação, sob orientação do Prof. Dr. Elvio Gilberto da Silva.

BAURU

2016

Mattos, Murilo Rosa de

M4446d

Gestão de conhecimento: desenvolvimento de um sistema baseado em ontologia / Murilo Rosa de Mattos. -- 2016.

107f. : il.

Orientador: Prof. Dr. Elvio Gilberto da Silva.

Trabalho de Conclusão de Curso (Graduação em Ciências da Computação) – Universidade do Sagrado Coração – Bauru – SP.

1. Gestão de conhecimento. 2. Ontologia. 3. Informação. 4. Inteligência Artificial. I. Silva, Elvio Gilberto da. II. Título

MURILO ROSA DE MATTOS

**GESTÃO DE CONHECIMENTO: DESENVOLVIMENTO DE UM SISTEMA
BASEADO EM ONTOLOGIA**

Trabalho de Conclusão de Curso apresentado ao centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de bacharel em Ciência da Computação, sob orientação do Prof. Dr. Elvio Gilberto da Silva.

Banca examinadora:

Prof. Dr. Elvio Gilberto da Silva
Universidade do Sagrado Coração

Prof. Me. Patrick Pedreira Silva
Universidade do Sagrado Coração

Prof^a. Ma. Daniela Luchesi
Universidade do Sagrado Coração

Bauru, 05 de dezembro de 2016.

AGRADECIMENTOS

Agradeço primeiramente aos meus pais por terem me proporcionado a criação e educação pela qual meu caráter foi formado, além de possibilitarem a minha jornada acadêmica na Universidade do Sagrado Coração.

Ao orientador do projeto e professor Dr. Elvio Gilberto da Silva pelo tempo dedicado, além de sua experiência e conhecimento sobre os assuntos tratados no projeto que certamente me guiaram no decorrer de todo o curso.

Ao corpo docente do curso de Ciência da Computação da Universidade do Sagrado Coração por toda a orientação e pelo apoio que tive desde o momento em que entrei na universidade.

Sou grato por todos os acadêmicos e profissionais de todas as áreas de tecnologia que contribuem com o crescimento e avanço tecnológico através do compartilhamento da informação e de suas descobertas.

RESUMO

A gestão do conhecimento é uma área de estudo que se encontra em ascensão graças ao avanço da tecnologia e do crescente investimento das empresas em recursos de otimização das rotinas corporativas. Ambos os fatores podem ser justificados pela viabilização e acessibilidade de projetos voltados ao empreendedorismo, tal como, pela atuação de empresas com espírito empreendedor que elevam o nível competitivo do mercado através de investimentos em tecnologia de ponta. Considerando o contexto apresentado e a oportunidade de criar um projeto de caráter inovador partindo do meio acadêmico, este trabalho tem como propósito possibilitar o preenchimento da lacuna das práticas da administração do conhecimento através da implementação de um sistema de gestão baseado em um advento da Inteligência Artificial que permite atribuir significado a uma entidade denominado como Ontologia fazendo uso das linguagens C#, Php e Mysql, assim como, dos protocolos Http e FTP, integrando um servidor Linux, que se comunica com a plataforma “.NET” para tornar possível a uma organização/empreendimento obter controle sobre o conhecimento trabalhado em seu cotidiano corporativo. Este projeto visa cumprir a premissa estabelecida partindo de uma aplicação desenvolvida em C# que integra uma ontologia criada com o objetivo de demonstrar o potencial de um sistema voltado à gestão do conhecimento a partir da classificação das informações abordadas diariamente pela empresa. Além de apresentar uma possibilidade simples e viável para a implementação de uma plataforma voltada a este aspecto, este trabalho incentiva a abordagem do conhecimento como recurso gerenciável através da implementação deste projeto. Como resultado, foi obtida uma taxa de 93.33% de precisão no processamento e classificação de conhecimento em formato de documentos textuais.

Palavras-chave: Gestão de conhecimento. Ontologia. Informação. Inteligência Artificial.

ABSTRACT

Knowledge Management is a field of study which is currently ascending thanks to the technological advancement and the increasing investment from companies on organization optimization routines. Both factors may be justified by the viability and accessibility of entrepreneurship projects as much as by the action of entrepreneurship-encouraging organizations which can increase the competitiveness in a certain business model through investments on cutting-edge technology. Considering the given context and the opportunity to create an innovative project coming from the academic environment, this project aims to fill a gap in the current practices of knowledge management through the implementation of an ontology-based system using C#, PHP and MySQL languages as much as the HTTP and FTP protocols embedded on a Linux server which should communicate with the .NET platform in order to enable a company to get control over the knowledge used in a daily basis in a certain corporation. This project aims to fulfill this premise using a C# developed application which integrates an ontology created with the objective of showing the potential of a system of knowledge management which can sort and organize information through a viable and simple application. Besides presenting this new perspective, this paper encourages a knowledge-based approach for a company management practices. As a result, the developed system holds a 93.33% accuracy rating on processing and sorting knowledge shaped as text documents.

Keywords: Knowledge Management. Ontology. Information. Artificial Intelligence.

LISTA DE ILUSTRAÇÕES

Figura 1 – Relação entre os termos dados, informação e conhecimento	17
Figura 2 – Sistema de informação aplicado na empresa	19
Figura 3 – Diagramas da UML	29
Figura 4 – Exemplo de Diagrama de Casos de Uso	30
Figura 5 – Exemplo de Diagrama de Classes	32
Figura 6 – Exemplo de Diagrama de Atividades	33
Figura 7 – Ciclo de aprimoramento de um software partindo de uma boa IHC	35
Figura 8 – Paradigma de linguagem estruturada	41
Figura 9 – Representação da CLR	43
Figura 10 – Compilação no ambiente .NET	43
Figura 11 – Estrutura de um programa em C#	45
Figura 12 – Transformações da sentença no processamento	55
Figura 13 – O teste de Turing	58
Figura 14 – Reconhecimento de textura pelo modelo bag-of-words	59
Figura 15 – Aplicações da ontologia	61
Figura 16 – aplicações utilizadas durante o desenvolvimento	66
Figura 17 – Cadeia de ações na interação com o sistema	69
Figura 18 – Diagrama de Atividades para inclusão de documentos	70
Figura 19 – Diagrama de Atividades para visualização de documentos	71
Figura 20 – Diagrama de Atividades para exclusão de documentos	72
Figura 21 – Diagrama de atividades para manter usuários	72
Figura 22 – Diagrama de Casos de Uso do sistema	73
Figura 23 – Diagrama de Classes do sistema	74
Figura 24 – Total de termos encontrados por categoria	76
Figura 25 – Termos selecionados por categoria	77
Figura 26 – Ocorrência de termos relevantes por categoria	78
Figura 27 – Termos selecionados para a categoria Comercial	79
Figura 28 – Termos selecionados para a categoria “Financeiro”	80
Figura 29.1 – Termos selecionados para a categoria “Rec. Humanos”	81
Figura 29.2 – Termos selecionados para a categoria “Rec. Humanos”	82
Figura 30 – Modelagem da tabela Users	83
Figura 31 – Modelagem da tabela Colaboradores	84
Figura 32 – Modelagem da tabela Arquivos	84
Figura 33 – Tela de upload de documentos	86
Figura 34 – Upload de um arquivo já existente no servidor com acesso superior	88
Figura 35 – Upload de um arquivo já existente no servidor sem acesso superior	88
Figura 36 – Tela de busca de documentos registrados	89
Figura 37 – Análise obtida a partir dos testes realizados	91

LISTA DE ABREVIATURAS E SIGLAS

CLR	– Common Language Runtime
ERP	– Enterprise Resource Planning
ES	– Engenharia de Software
IA	– Inteligência Artificial
IHC	– Interface Homem-Computador
MSDN	– Microsoft Developer Network
PLN	– Processamento de Linguas Naturais
POO	– Programação Orientada a Objetos
SGBD	– Sistema de Gerenciamento de Banco de Dados
SI	– Sistema(s) de Informação
SQL	– Structured Query Language
TS	– Teste de Software
UML	– Unified Model Language
PHP	– PHP: Hypertext Processor
LAMP	– Linux, Apache, MySQL, PHP
FTP	– File Transfer Protocol

SUMÁRIO

1	INTRODUÇÃO	13
2	OBJETIVOS	16
2.1	OBJETIVO GERAL	16
2.2	OBJETIVOS ESPECÍFICOS	16
3	DADOS, INFORMAÇÃO E CONHECIMENTO	17
3.1	SISTEMAS DE INFORMAÇÃO (SI)	18
3.2	SISTEMAS INTEGRADOS DE RECURSOS EMPRESARIAIS (ERP)	19
3.3	GESTÃO DO CONHECIMENTO.....	21
4	SOFTWARE	23
4.1	CARACTERÍSTICAS DE UM SOFTWARE	23
4.2	TIPOS DE SOFTWARE	24
4.3	ENGENHARIA DE SOFTWARE (ES)	26
4.4	A LINGUAGEM DE MODELAGEM UNIFICADA (UML)	27
4.4.1	Os diagramas de modelagem UML	28
4.4.1.1	O diagrama de casos de uso.....	29
4.4.1.2	O diagrama de classes.....	31
4.4.1.3	O diagrama de atividades.....	32
4.5	ANÁLISE E LEVANTAMENTO DE REQUISITOS.....	33
4.6	A INTERAÇÃO HOMEM-COMPUTADOR (IHC).....	34
4.7	TESTE DE SOFTWARE (TS).....	36
4.7.1	Tipos de teste de software	37
5	ALGORITMO	39
5.1	O PARADIGMA DA ORIENTAÇÃO A OBJETOS	40
5.1.1	Embasamento para o uso do paradigma atual	41
5.2	A PLATAFORMA .NET	42
5.2.1	A Linguagem de programação C#	44
5.3	A LINGUAGEM PHP	45
6	O BANCO DE DADOS ESTRUTURADO	46
6.1	O BANCO DE DADOS RELACIONAL	47
6.2	A LINGUAGEM SQL	48

6.3	SISTEMAS DE GERENCIAMENTO DE BANCOS DE DADOS (SGBD)	48
6.4	O BANCO DE DADOS MYSQL	49
7	INTELIGÊNCIA ARTIFICIAL (IA)	51
7.1	A RELAÇÃO COM A INTELIGÊNCIA HUMANA	52
7.2	HISTÓRIA DA INTELIGÊNCIA ARTIFICIAL	52
7.3	APLICAÇÕES DA INTELIGÊNCIA ARTIFICIAL	54
7.4	PROCESSAMENTO DE LINGUAS NATURAIS (PLN)	55
7.4.1	O teste de Turing	57
7.4.2	O modelo bag-of-words	59
7.5	ONTOLOGIA	60
7.6	TAXONOMIA	62
8	TRABALHOS CORRELATOS	64
9	METODOLOGIA	66
9.1	FERRAMENTAS UTILIZADAS	66
9.1.1	Hardware	66
9.1.2	Software	67
9.2	ESTRUTURA DO SERVIDOR	69
9.3	LINGUAGENS DE PROGRAMAÇÃO E MODELAGEM	70
9.4	MODELAGEM DO SISTEMA	71
9.4.1	Diagramas de Atividades	71
9.4.2	Diagrama de Casos de Uso	75
9.4.3	Diagrama de classes	75
9.5	ELABORAÇÃO DA ONTOLOGIA	77
9.5.1	Estudo exploratório de modelos disponíveis para implementação	77
9.5.2	Extração de palavras pertinentes ao contexto	77
9.5.3	Avaliação e seleção dos termos levantados	78
9.5.4	Implementação no sistema em C#	84
9.6	ARMAZENAMENTO E ORGANIZAÇÃO DO CONHECIMENTO	85
9.7	TESTES DO SISTEMA	87
10	ELABORAÇÃO DO SOFTWARE	88
10.1	TELAS DO SISTEMAS	88
10.1.1	Upload de documentos	88

10.1.2 Busca de documentos.....	91
11 VALIDAÇÃO DOS RESULTADOS OBTIDOS.....	93
12 TRABALHOS FUTUROS.....	94
13 CONSIDERAÇÕES FINAIS.....	95
REFERÊNCIAS.....	96
APÊNDICE A – TELAS DO SISTEMA.....	104

1 INTRODUÇÃO

A gestão de conhecimento aplicada nas empresas deve ser vista como um meio de consolidar, organizar, refinar e extrair o máximo de valor possível de todo o conjunto de informações que é tratado em seu cotidiano com o propósito de aumentar a vantagem competitiva em função do cenário brasileiro comercial.

É evidente, e até mesmo compreensível, que as empresas hoje em dia ainda apresentem resistência a investir em tecnologia, devido ao fato de que muitas delas carregam casos de más experiências como resultado de tal investimento. No entanto, grande parte dos incidentes são justificáveis por uma má aplicação de recursos e de gerenciamento no momento da implementação, o que gera como resultado final o pré-conceito em acreditar na tecnologia devidamente.

Sendo assim, é importante ressaltar que a tecnologia por si só não é capaz de mudar a forma como as coisas funcionam dentro de uma empresa. É necessário que haja planejamento e administração dos recursos computacionais, assim como é feito em qualquer outro setor corporativo em momentos de inovação, para que um resultado positivo seja alcançado.

A gestão de conhecimento é uma das áreas pouco exploradas pelas empresas brasileiras, seja pela falta de domínio sobre o assunto, o medo dos empreendedores de investir ou pela negligência e falta de informação sobre o próprio conteúdo que é contido e trabalhado no cotidiano corporativo. É, portanto, uma das áreas com grande potencial de exploração, tanto com o propósito de avanço acadêmico-científico como também o de melhorar a qualidade da informação detida pelas empresas.

O conhecimento retido por uma empresa é produto de seu investimento no decorrer dos anos – seja esse investimento representado na forma de capacitação de funcionários, contratações, materiais ou a experiência individual de cada colaborador acumulada no decorrer de sua jornada profissional. É comum que esses colaboradores sintam-se donos de todo o conhecimento que acumularam e que isso seja aceito como algo normal entre os outros membros de uma equipe. No entanto, esse tipo de mentalidade é altamente prejudicial a todas as partes envolvidas, pois se por um lado a empresa se torna “refém” do monopólio de informação dos colaboradores mais

capacitados, por outro os próprios funcionários deixam de crescer profissionalmente compartilhando informações entre eles.

Levando em consideração o contexto retratado, esta pesquisa visa introduzir o empreendedor ao meio da tecnologia do conhecimento, através da aplicação de um sistema de classificação e gerenciamento de informação, o qual foi desenvolvido fazendo uso de uma ontologia como recurso de Inteligência Artificial, e tem o propósito final de apresentar ao mercado corporativo brasileiro uma forma de destacar-se dos demais utilizando um sistema que o permite aplicar a administração do conhecimento nos moldes previamente abordados.

O conceito de gestão de conhecimento ainda é relativamente novo, embora bastante explorado, no contexto brasileiro, o que resulta em um desperdício do potencial e da eficiência nas empresas que não tem ciência sobre a capacidade contida em gerenciar a informação no âmbito corporativo. Sendo assim, surge a oportunidade de abordar e explorar a gestão de conhecimento em um dos meios em que ela pode ser tratada atualmente, a ciência da computação.

Partindo do ponto de vista da ciência da computação, a gestão do conhecimento pode ser encarada como um dos muitos recursos gerenciáveis disponíveis no mercado. Em particular, trata-se de um recurso de extrema importância e pouco tratado na área da tecnologia, o que evidencia a possibilidade de crescimento disponível ao vinculá-lo a outros recursos como a Inteligência Artificial e o gerenciamento de dados através de sistemas dedicados.

No atual estado em que se encontram as empresas, o investimento corporativo em conhecimento e informação pode acabar resultando em cenários de prejuízo, seja pela negligência em como esses documentos são tratados ou pela falta de praticidade em criar uma lógica corporativa para cuidar do patrimônio intelectual da empresa, o que em suma pode resultar na resistência da diretoria de negócio em abraçar a tecnologia e o conhecimento, limitando-a por um conjunto de más experiências agregadas.

A miopia sobre o problema e a ignorância corporativa não são desculpas plausíveis para que uma empresa se restrinja de ser melhor e se destacar do resto do mercado. Com o propósito de minimizar tudo isso e de auxiliá-la a extrair o máximo de valor do conjunto de informações nela contida, foi desenvolvido um sistema de gestão e

classificação de informação que busca aumentar a lucratividade intelectual através do gerenciamento de conhecimento que foi acumulado em determinada instituição ao longo do tempo.

2 OBJETIVOS

A seguir são apresentados o objetivo geral e os específicos que norteiam este trabalho.

2.1 OBJETIVO GERAL

Desenvolver um sistema de classificação e gerenciamento de conhecimento baseado em ontologia.

2.2 OBJETIVOS ESPECÍFICOS

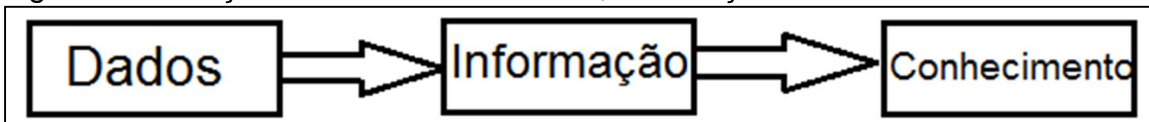
- a) Levantar os requisitos necessários para o desenvolvimento do projeto;
- b) Investigar e compreender os aspectos e recursos essenciais para o desenvolvimento do sistema;
- c) Planejar e analisar as características do projeto para implementação;
- d) Desenvolver a ontologia que será utilizada;
- e) Modelar o sistema de informação;
- f) Desenvolver o sistema de informação;
- g) Testar e avaliar o funcionamento do sistema.

3 DADOS, INFORMAÇÃO E CONHECIMENTO

É comum que as pessoas de fora do meio computacional não compreendam a diferença entre os dois termos. Para isso, Laudon e Laudon (2011) explicam que a informação pode ser definida como um conjunto de dados que é apresentado com significado e/ou utilidade, enquanto dados podem ser considerados como sequências de fatos sem qualquer tipo de processamento.

Essas definições dadas pelos autores deixam muito clara a diferença entre os dois termos. Por um lado, a informação pode ser considerada um conjunto de dados dentro de um determinado contexto, enquanto um dado é um fragmento de conhecimento qualquer sem que tenha tido qualquer sinal de esforço para compreendê-lo. É possível visualizar a relação entre as três áreas através da Figura 1.

Figura 1 – Relação entre os termos dados, informação e conhecimento.



Fonte: elaborada pelo autor.

Ackoff (1989, citado por BELLINGER; CASTRO; MILS, 2004) define dados como algo cru; que simplesmente existe e que não há qualquer significado além de sua existência (ou sequer em sua própria existência), sejam esses dados utilizáveis ou não.

O autor completa definindo informação como um conjunto de dados a partir do qual foi dado significado através de uma forma relacional. Não existe necessariamente utilidade para a informação, a não ser que um mecanismo qualquer de gerenciamento atribua um sentido a ela.

Por último, Ackoff aborda conhecimento como um conjunto apropriado de informação, tal que esse conjunto é propositalmente útil. Conhecimento é, portanto, um processo determinístico que necessariamente requer que um agrupamento de informações possua um significado.

3.1 SISTEMAS DE INFORMAÇÃO (SI)

Sistemas de informação são *softwares* que, atualmente, dominam a área comercial em empresas de qualquer porte, possibilitando o aumento da vantagem competitiva através da automatização de processos e gerenciamento de informações.

Um sistema de informação pode ser definido tecnicamente como um conjunto de componentes inter-relacionados que coletam (ou recuperam), processam, armazenam e distribuem informações destinadas a apoiar a tomada de decisões, a coordenação e o controle de uma organização. Além de dar apoio à tomada de decisões, à coordenação e ao controle, esses sistemas também auxiliam os gerentes e trabalhadores a analisar problemas, visualizar assuntos complexos e criar novos produtos. (LAUDON, J.; LAUDON, K., 2009, p. 9).

Tendo a definição dos autores em mente, é possível compreender a influência e o potencial de um sistema de informação em uma empresa. Além dos benefícios claros que resultam em um aumento na vantagem competitiva, também é necessário ressaltar que isso acaba reforçando à companhia a importância de modernização.

Guimarães e Évora (2004) abordam ainda outro aspecto dos sistemas de informação que ainda não foi tratado. Segundo esses autores que consideram a tomada de decisão como uma função que caracteriza o desempenho da gerência, “independente do aspecto da decisão, esta atitude deve ser fruto de um processo sistematizado”. (GUIMARÃES; ÉVORA, 2004, p. 73).

Deste modo, compreende-se que um sistema de informação possui uma gama de ferramentas e utilidades muito mais do que pode se esperar do ponto de vista de uma pessoa nova no assunto. Isso pode ser confirmado pelo fato de que tudo que acontece dentro de uma empresa gera informação e toda informação é passível de ser gerenciada.

Segundo Filho (1994, p. 34), a Figura 2 evidencia a relação entre os componentes que fazem parte de um sistema de informação, além da relação entre eles. “[...] Essa figura enfatiza que o ponto focal para o entendimento da natureza do S.I. são as práticas de trabalho, e não as tecnologias de informação, como pode parecer à primeira vista.”, completa o autor.

Figura 2 – Sistema de informação aplicado na empresa.



Fonte: Filho (1994, p. 34).

É interessante ressaltar que a Figura 2 demonstra as práticas de trabalho como parte de um sistema de informação, pois essa visão, ainda nos dias de hoje, pode não ser necessariamente a forma como um SI é visto pelos olhos de empresários, investidores e até mesmo colaboradores inexperientes no assunto.

As práticas de trabalho consistem nos métodos utilizados pelos recursos humanos para desempenhar suas atividades no S.I., incluindo não somente os procedimentos descritos pelos manuais de operação, mas também as linhas de ação nas quais os recursos humanos se coordenam, se comunicam e tomam decisões, realizam negócios ou serviços e desempenham outras tarefas. (FILHO, 1994, p. 35).

3.2 SISTEMAS INTEGRADOS DE RECURSOS EMPRESARIAIS (ERP)

Os ERPs ficaram conhecidos no mundo administrativo por serem flexíveis e poderem ser adequados a qualquer empresa que deseja automatizar processos. Isso se dá ao fato de que os sistemas geralmente permitem a implementação de módulos específicos para tratar de cada área da empresa, o que acaba por gerar um mesmo ambiente em que os funcionários já estão acostumados a utilizar com uma gama ilimitada de funcionalidades.

[...] no final da década de 1980, um novo conceito de *software* foi apresentado com a finalidade de abranger todas as áreas da organização, agrupando as funcionalidades dos diversos sistemas de *software* “isolados” em um único sistema que abrangesse desde folha de pagamento, contabilidade, financeiro, compras, vendas, até os processos industriais. Assim surgiram os primeiros sistemas ERP, com o grande diferencial de substituir os diversos sistemas isolados, eliminando redundância de dados e com funcionalidades abrangentes que cobrem diversas áreas da organização de forma orientada a processos. (CORREA; SPINOLA, 2015).

Os sistemas de gestão de recursos empresariais não necessariamente surgiram a partir de uma necessidade que as empresas da época tinham, mas sim, como tratado por CORREA e SPINOLA (2015), tratava-se de uma área que foi desbravada e que teve como produto um novo conceito de *software* completamente inovador que oferecia diversos recursos de automatização com o propósito de aumentar a vantagem competitiva. Nesse contexto, portanto, pode-se dizer que a tecnologia gerou a necessidade.

Segundo Chopra e Meindl (2003 citado por PADILHA; MARINS, 2005, p. 104): “Os sistemas ERP fornecem rastreamento e visibilidade global da informação de qualquer parte da empresa e de sua cadeia de suprimento, o que possibilita decisões inteligentes.”

Segundo Hossain, Patrick e Rashid (2002), sistemas de planejamento de recursos empresariais são *softwares* para o gerenciamento de negócio que abrangem áreas como o planejamento, manufatura, vendas, marketing, distribuição, financeiro, recursos humanos e, entre outros, gerenciamento de projetos com uma arquitetura que facilita uma integração transparente dos módulos, o que oferece um fluxo de informação entre todas as funcionalidades dentro da companhia de forma visivelmente consistente.

Hossain, Patrick e Rashid (2002) completam afirmando que esse tipo de sistema, de um modo geral, deve melhorar tanto a estrutura como o produto final de uma empresa simultaneamente. As organizações costumam escolher pela implementação de um ERP por motivos tangíveis e intangíveis, além de razões estratégicas.

Com isso, pode ser concluído que embora os sistemas de recursos empresariais de fato aumentam a produtividade e a vantagem competitiva no meio em que são incluídos, e de acordo com muitos autores essa melhoria não acontece da noite para o

dia como é esperado pelos investidores. Alguns desses escritores até mesmo colocam essas expectativas intangíveis como um dos obstáculos na etapa de implementação do sistema.

3.3 GESTÃO DO CONHECIMENTO

A gestão de conhecimento pode ser encarada como um aprofundamento da gestão da informação vista anteriormente. Tendo em vista a necessidade de continuar sempre em um estado de adaptação e evolução no cenário competitivo, é importante que uma empresa sempre busque novos meios de transformar-se em uma organização melhor em contraste à concorrência. A gestão do conhecimento permite que isso seja possível através de um conjunto de práticas que podem ser aplicadas a qualquer organização.

Em uma economia onde a única certeza é a incerteza, a fonte certa e vantagem competitiva duradoura é o conhecimento. Quando os mercados transformam-se, as tecnologias proliferam, os competidores multiplicam-se e os produtos tornam-se obsoletos quase do dia para a noite, as empresas bem-sucedidas são as que criam consistentemente novos conhecimentos, disseminam-no amplamente pela organização e o incorporam rapidamente em novas tecnologias e produtos. (TAKEUCHI; NONAKA, 2008, p. 39).

Tendo o ponto de vista de Takeuchi e Nonaka (2008) em mente, é possível concluir que a propriedade intelectual, apesar de não ser o foco principal de muitas empresas, é a única constante em uma organização, mesmo que os empresários envolvidos no cotidiano da companhia não saibam disso.

Gestão do conhecimento é um modelo interdisciplinar de negócio, emerso recentemente, que lida com todos os aspectos do conhecimento dentro do contexto da empresa, inclusive criação, codificação, compartilhamento, aprendizado e inovação do conhecimento (PONZI, 2002, p. 268, citado por ROSSETTI, 2009, p. 26, tradução do autor).

Sendo assim, é possível compreender a gestão do conhecimento como uma vertente dos sistemas de informação que abrange uma área que ainda não tinha sido

explorada pelas empresas: o conhecimento como propriedade passível de gerenciamento em uma empresa.

Segundo Cardoso e Machado (2008), essa área abrange todos os aspectos que manipulam o conhecimento em determinada forma e a grande quantidade destes aspectos, junto ao grande volume de dados relevantes, acabam por gerar a necessidade de que exista algum tipo de tecnologia de informação envolvida no processo de gestão.

Segundo Figueiredo (2006), embora a gestão do conhecimento seja uma necessidade no cotidiano das organizações, este assunto ainda é bastante negligenciado, pois há falta de compreensão ampla sobre o assunto, além de que as empresas não vão além da gerencia dos recursos tangíveis da organização.

De acordo com Santiago Junior (2004), é necessário que as empresas compreendam o quanto antes que o conhecimento passou a ser um ativo indispensável, além de ser a matéria-prima com a qual todas as organizações lidam no cotidiano. O autor completa seu ponto de vista argumentando que o conhecimento é tão crucial tal que este pode ser considerado um dos fatores de sobrevivência de uma empresa.

Gerir os conhecimentos em uma empresa é, antes de mais nada, tornar comum as experiências de cada um. É incompreensível que, ainda hoje, muitas empresas ainda valorizem o individualismo em detrimento do trabalho cooperativo. Este é, sem sombra de dúvida, um dos principais obstáculos ao sucesso da Gestão do Conhecimento. Se a organização não incentiva o compartilhamento de informações, o funcionário se sente dono do que sabe. (ZAIDAN, 2010?)

4 SOFTWARE

De acordo com Sommerville (2012), *softwares* são programas de computador e documentação associada, sejam esses programas criados para um indivíduo ou para o mercado de *software* de forma geral. Também é importante ressaltar que um *software* considerado bom deve providenciar ao usuário funcionalidade e desempenho, além de ser confiável e fácil de manter e usar.

Pressman (2000) descreve *software* como programas que quando executados desempenham determinada desejada função e performance. O autor completa enfatizando que se trata de estruturas de dados que permitem a manipulação de informação a partir de determinado programa.

É interessante apontar que ambas as definições dadas pelos autores fazem referência às características de um *software*, destacando que um bom desempenho é essencial para que um *software* seja definido como tal. Pode-se concluir, portanto, que tais características são as que permitem tal classificação.

4.1 CARACTERÍSTICAS DE UM SOFTWARE

De acordo com Sommerville (2012) existem quatro características essenciais para que um *software* seja considerado bom. São essas: manutenibilidade, confiança e proteção, eficiência e aceitabilidade.

- a) **Manutenibilidade:** Trata-se do fator de manutenção de um *software*. De acordo com o autor, todo programa deve ter potencial de evoluir de acordo com a necessidade do usuário através de manutenção em sua estrutura. Assim, fica clara a importância da capacidade de crescimento para que um sistema resista ao tempo;
- b) **Confiança e proteção:** Estes são os aspectos que definem a segurança de um bom *software*. De acordo com o autor, um *software* confiável é aquele que não causa nenhum tipo de prejuízo em caso de falhas ou tentativas de intrusão.

- c) **Eficiência:** Esta característica é definida pelo bom uso dos recursos solicitados para o funcionamento do *software*, sem que haja desperdício de memória e tempo de processamento;
- d) **Aceitabilidade:** O *software*, ao ser desenvolvido, deve levar em consideração os outros recursos e componentes que são utilizados no cotidiano do usuário para o qual ele é designado. Assim, um bom *software* tem que ser compatível e compreensível para tal usuário.

Essas características podem ser observadas em todos os sistemas de *software* que são utilizados no dia-a-dia e que são conhecidos pela população em geral. Sendo esses aspectos intrínsecos dos programas consolidados na cultura de informação atual, pode-se afirmar que é primordial para qualquer *software* bem sucedido que cubra essas quatro características.

4.2 TIPOS DE SOFTWARE

Pressman (2000) define que um *software* pode ser aplicado em qualquer contexto em que uma determinada quantidade rotinas possa ser estabelecida, o que gera a necessidade de especificação ao desenvolver um novo programa, dada a abrangência em que pode ser contextualizado. O autor apresenta os seguintes tipos de *software* como exemplo de aplicações em potencial:

- a) **Software de sistema:** são coleções de *software* que servem como intermediário entre outros programas e a própria máquina. É característica desse tipo de *software* a interação com os componentes físicos de um computador através do gerenciamento dos recursos computacionais;
- b) **Software de tempo real:** Trata-se dos programas que tem a capacidade de monitoramento, análise e controle de eventos no mundo. O autor define como componentes desse tipo de *software* um sistema de coleta de dados e formatação desses dados para a linguagem compreensível do programa, um sistema de análise dos dados coletados, um componente de controle que

permite que o programa responda ao mundo real e um componente de monitoramento que gerencia todos os outros;

- c) **Software de negócio:** O autor afirma que sistemas de processamento de informações empresariais compõem a maior de todas as áreas de aplicação de *software*. Esse tipo de *software* pode desempenhar incontáveis funções dentro de uma empresa, incluindo reestruturação de informação para facilitar a compreensão dos usuários, gerenciamento de recursos, conversão de dados e agentes de tomada de decisão;
- d) **Software científico e de engenharia:** De acordo com o autor, esse modelo de *software* é caracterizado pelos algoritmos de processamento numérico. Esse tipo de programa está presente em todo tipo de contexto acadêmico moderno, por exemplo, a astronomia e biologia molecular;
- e) **Sistemas embarcados:** Esse tipo de *software* está presente no cotidiano mais que qualquer outro, pois se trata dos sistemas de controle de aparelhos e automóveis que possuem algum tipo de interação digital com o usuário, como computadores de bordo e eletrodomésticos;
- f) **Software de uso pessoal:** O autor aponta para esse tipo de *software* como aquele que tem surgido desde a década de 90 como sendo o mais abrangente e variado entre todos os outros tipos, pois se trata de *softwares* que abordam todo tipo de necessidade pessoal, incluindo entretenimento, computação gráfica, multimídia e processadores de texto;
- g) **Software baseado em web:** Esse tipo de *software* possui como principal característica uma quantidade ilimitada de conteúdo, pois sua funcionalidade é de capturar dados através da rede para que estes sejam mostrados ao usuário;
- h) **Software de Inteligência Artificial:** Segundo o autor, esse tipo de *software* tem como funcionalidade a resolução de problemas complexos que não poderiam ser resolvidos através do sistema computacional convencional. Sistemas especialistas, redes neurais e sistemas de reconhecimento de padrões são exemplos da aplicação de *softwares* de Inteligência Artificial.

Sommerville (2012) ainda se refere a outros tipos de aplicação, como sistemas para modelagem e simulação, sistemas de coleta de dados, sistemas baseados em transações e sistemas de sistemas.

4.3 ENGENHARIA DE SOFTWARE (ES)

A engenharia de *software* é uma disciplina essencial para o estabelecimento de um conjunto de boas práticas que viabilizam a distribuição e comercialização de *softwares* empresariais e de uso pessoal, pois antes de ser explorada a fundo, não existiam garantias e segurança de que o acordo entre usuário e desenvolvedor seria mantido com sucesso.

Segundo Sommerville (2012, p. 5), “Engenharia de software é uma disciplina de engenharia cujo foco está em todos os aspectos da produção de *software*, desde os estágios iniciais da especificação do sistema até sua manutenção, quando o sistema já está sendo usado.”.

A definição dada é importante para compreender o papel da engenharia de *software* na atualidade, pois demonstra que a aplicação dessa disciplina ocorre em todas as fases do desenvolvimento – e de pós-desenvolvimento – de um programa ou sistema.

A qualidade da educação em ES pode contribuir significativamente à melhoria do estado da arte do desenvolvimento de software e auxiliar a solução de alguns problemas tradicionais e crises relacionadas com as práticas da indústria de software. (GIBBS, 1994, citado por ALBUQUERQUE et al., 2009).

De acordo os autores, o desenvolvimento tecnológico no mundo e a presença do *software* cada vez mais intrínseca a todo tipo de empresa competitiva no mercado, são aspectos que justificam o aumento do investimento nesse setor, tal como o aumento da qualidade do produto que é desenvolvido.

Os autores complementam esse ponto de vista utilizando como parâmetro a produção de *software* pré-engenharia de *software* em um período em que o programa poderia ser desenvolvido a partir de um simples levantamento de requisitos, o que evidencia o peso da disciplina hoje em dia. Essa perspectiva sobre o passado pode ser

levada em consideração também com o propósito de demonstrar o avanço da engenharia de *software* com o passar dos anos.

4.4 A LINGUAGEM DE MODELAGEM UNIFICADA (UML)

A linguagem de modelagem unificada pode ser considerada como uma peça fundamental para a boa engenharia de *software*, pois se trata de um conjunto de diagramas de modelagem que expressam a intenção da empresa desenvolvedora em função do que será desenvolvido. Desta forma, contribui para uma boa ES no que se diz respeito ao planejamento pré-desenvolvimento.

De acordo com Guedes (2004), a UML é uma linguagem visual que almeja modelar sistemas computacionais através de vários tipos de diagramas que compõe a linguagem. A linguagem de modelagem, no entanto, possui independência da linguagem de programação e plataforma de desenvolvimento que serão utilizados no decorrer do projeto devido ao fato de que esses não são pontos considerados relevantes pela UML.

A UML (Unified Modeling Language) é uma linguagem-padrão para a elaboração da estrutura de projetos de software. Ela poderá ser empregada para a visualização, a especificação, a construção e a documentação de artefatos que façam uso de sistemas complexos de software. (BROOCH; JACOBSON; RUMBAUGH, 2006, p. 13).

A visão dos autores sobre a definição da UML é bastante pertinente por demonstrar que existe uma variedade de funcionalidades que pode ser utilizada com a linguagem em determinado contexto, além da importância de se fazer uso deste recurso sempre que houver a necessidade.

Guedes (2004) estabelece uma analogia entre a importância da UML com o processo que é utilizado por um engenheiro civil ao projetar um prédio de vários andares, apresentando o fato de que um edifício simplesmente não pode ser modelado de cabeça.

Da mesma forma que o autor demonstra a importância organizacional sobre a linguagem de modelagem unificada, também é possível levantar outros fatores

decisivos que colaboram com esse ponto de vista, sendo um deles a possibilidade de se levantar estimativas de tempo e recursos que serão utilizados no decorrer do projeto partindo de seus modelos criados a partir do levantamento de requisitos.

4.4.1 Os diagramas de modelagem UML

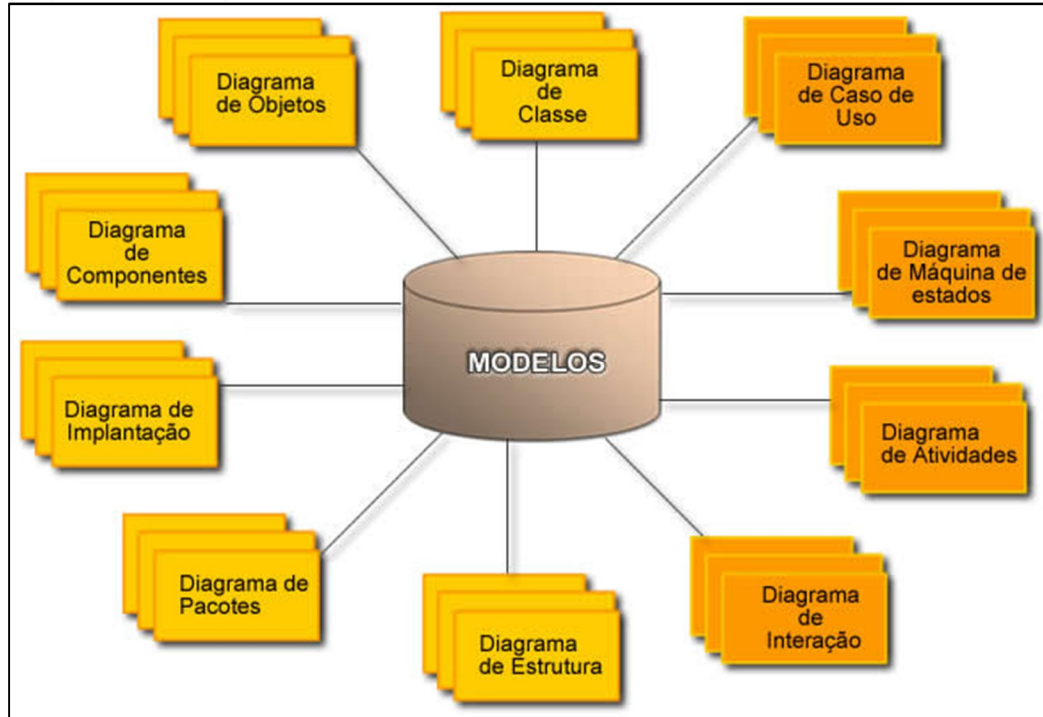
Existem 10 tipos de diagramas diferentes que compõe a linguagem de modelagem unificada, sendo cada um deles independente de qualquer outro, além de expressar uma função exclusiva e insubstituível pelos demais.

Os Diagramas da UML estão divididos em Estruturais e Comportamentais, conforme ilustra a Figura 3.

- a) **Diagramas Estruturais:** de Classe; de Objeto; de Componentes; de implantação; de Pacotes; de Estrutura.
- b) **Diagramas Comportamentais:** de Caso de Uso (Use Case); de Máquina de Estados; de Atividades; de Interação: Dividem-se em: de Sequência; Geral interação; de comunicação; de tempo.

Autores como Guedes (2004), Fowler (2005) e Bezerra, E. (2015) apresentam os principais deles como sendo: Diagrama de Caso de Uso, Diagrama de Classes, Diagrama de Objetos, Diagrama de Sequência, Diagrama de Atividades, Diagrama de Estado, Diagrama de Componentes e Diagrama de Pacotes.

Figura 3 – Diagramas da UML.



Fonte: Martinez (2006-2016).

Alguns autores como Fowler (2005) ainda categorizam esses diagramas como sendo de estrutura, comportamento e interações. O autor ressalta que essa distinção é importante por deixar claro que não existe uma necessidade de se fazer uso de todos os tipos de diagramas disponíveis para todo e qualquer projeto.

4.4.1.1 O diagrama de casos de uso

O Diagrama de Casos de Uso pode ser considerado como um dos mais famosos entre todos os tipos, por geralmente ser utilizado em todo tipo de projeto no decorrer de seu desenvolvimento.

Guedes (2004) afirma que sua aplicação normalmente é feita para que seja consultada nas etapas de levantamento de requisitos, mas que também pode ser utilizado durante todas as fases do projeto.

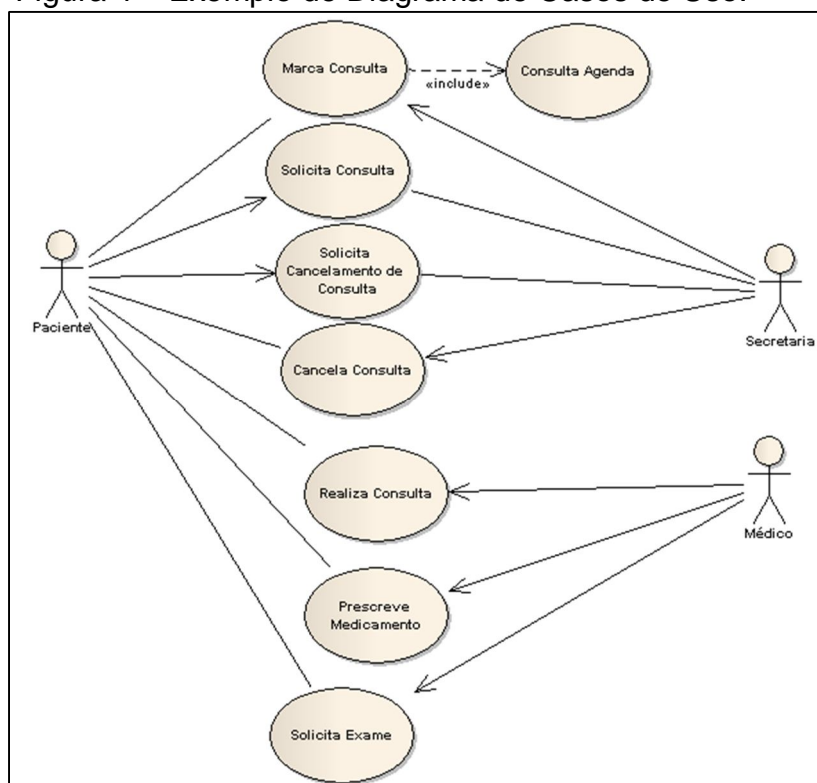
De acordo com Bezerra, E. (2015), a visão de casos de uso é criada nas etapas iniciais de um projeto e o direciona partindo de seu ponto de vista. O recurso é utilizado

com o propósito de enxergar os componentes com uma visão externa, tal como suas interações e agentes que causam a interação.

Ambos os autores apontam para o fato de que o Diagrama de Casos de Uso é muito importante nas etapas iniciais do projeto, além do fato de servir de apoio e alinhamento para que outros diagramas sejam desenvolvidos. É importante notar que a visão externa providenciada por este modelo de diagrama tem a capacidade de demonstrar todas as interações e consequências sobre cada tentativa de tomada de decisão por parte do “ator” (indivíduo qualquer pertinente ao sistema representado pelo diagrama).

A Figura 4 ilustra um exemplo de Diagrama de Casos de Uso.

Figura 4 – Exemplo de Diagrama de Casos de Uso.



Fonte: Elaborada pelo autor.

4.4.1.2 O diagrama de classes

O Diagrama de Classes é considerado por alguns autores como sendo o mais utilizado e o mais importante da UML. Isso se dá ao fato de que ele pode servir como apoio para ambos - o desenvolvimento do projeto e dos demais diagramas.

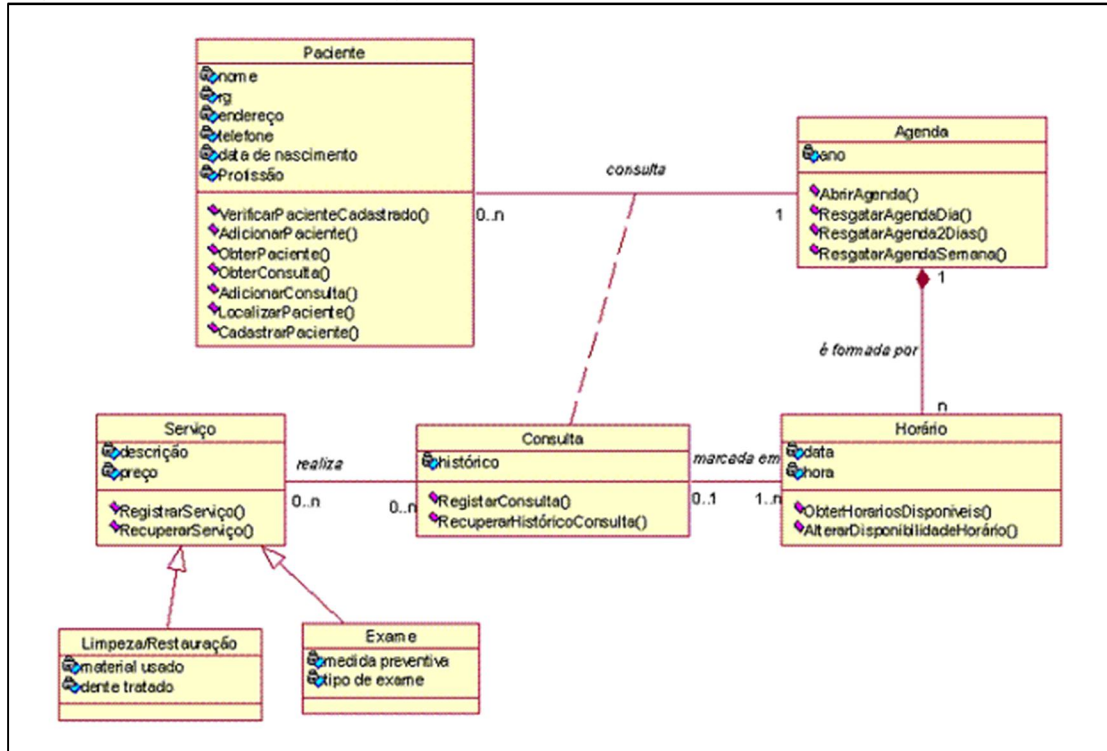
Guedes (2004, p. 27) define que “Como o próprio nome diz, define a estrutura das classes utilizadas pelo sistema, determinando os atributos e métodos possuídos por cada classe, além de estabelecer como as classes se relacionam e trocam informações entre si.”.

A definição do autor deixa claro que, embora a UML possa ser representada sobre qualquer linguagem de programação com a qual o sistema será desenvolvido, sem que haja necessidade de qualquer adaptação para implementação, ainda é necessário que haja algum conhecimento sobre o que será programado e sua estrutura.

Melo (2010) ainda alerta para o fato de que este diagrama pode ser desenvolvido após a extração dos requisitos para, que então sejam modeladas as classes do sistema.

A Figura 5 ilustra um exemplo de Diagrama de Classes.

Figura 5 – Exemplo de Diagrama de Classes.



Fonte: Elaborada pelo autor.

4.4.1.3 O diagrama de atividades

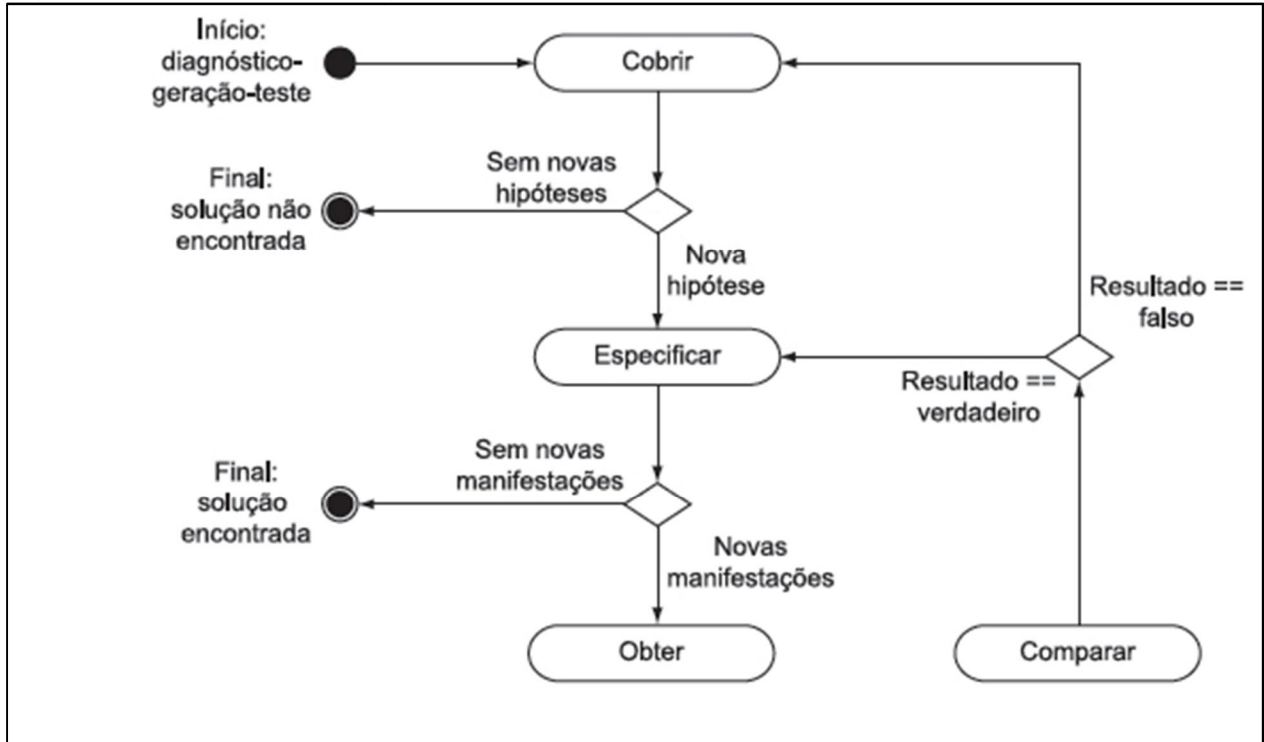
O Diagrama de Atividades se comporta como um fluxograma que representa detalhadamente as fases para que uma determinada atividade aconteça dentro do contexto de um sistema.

De acordo com o portal Microsoft Developer Network (MSDN), o Diagrama de Atividades é um fluxo de atividades que compõe uma atividade. No entanto, também é possível descrever fluxos de dados e esquemas mais complexos utilizando-o, embora não seja uma prática comum.

Guedes (2004) define que o diagrama de atividades tem como principal foco descrever os passos de execução para que uma determinada atividade seja concluída. O autor também complementa que, embora não seja prática comum da engenharia de *software*, também é possível de se utilizá-lo para representar o sistema como um todo, indo além da premissa de materializar uma classe de atividade complexa.

A Figura 6 ilustra um exemplo de Diagrama de Atividades.

Figura 6 – Exemplo de Diagrama de Atividades.



Fonte: Elaborada pelo autor.

4.5 ANÁLISE E LEVANTAMENTO DE REQUISITOS

Embora seja uma visão contra intuitiva, a fase de levantamento de requisitos vai muito além de obter as informações necessárias do ponto de vista do usuário para que o sistema seja projetado conforme as expectativas.

De acordo com Bezerra, E. (2015), o objetivo de se realizar um levantamento dos requisitos é alinhar as metas dos desenvolvedores com os interesses dos usuários, para que então ambas as partes possam definir as necessidades que serão satisfeitas pelo sistema que será desenvolvido.

Assim, fica claro que essa etapa da construção de um projeto pode ser considerada como uma das primeiras que deve ser abordada, alinhando a visão do desenvolvedor com o que é esperado pelos clientes na entrega do produto final.

Guedes (2004) se refere ao levantamento de requisitos como a fase em que o engenheiro de *software* tenta se aproximar da mentalidade do cliente, dessa forma compreendendo-o. De acordo com o autor, esse levantamento deve ser realizado a

partir de uma quantia indeterminada de entrevistas nas quais a equipe de desenvolvimento deve auxiliar seus clientes com a definição de suas necessidades.

O autor ainda aponta para o fato de que a análise dos requisitos levantados é um subprocesso de extrema importância, pois deve ser levado em consideração todos os apontamentos feitos pelos usuários com o propósito de examinar, e verificar se não houve falhas no decorrer da comunicação nessa etapa inicial.

Outras questões de peso que surgem durante a fase de levantamento de requisitos são a importância de uma boa comunicação entre todas as partes envolvidas no projeto, bem como, o subjetivismo de alguns critérios que são levantados pelos clientes. Todos esses aspectos devem necessariamente ser tratados antes de que qualquer etapa de desenvolvimento prático comece a ser realizada sobre o que foi levantado.

4.6 A INTERAÇÃO HOMEM-COMPUTADOR (IHC)

A área de estudo multidisciplinar IHC é, hoje em dia, fator definitivo sobre o qual é dada aprovação ou desaprovação de um *software* ou sistema desenvolvido para um determinado cliente.

Barbosa e Silva (2010) relatam que a disciplina tem como fundamento outras áreas como *design*, ergonomia, linguística e semiótica, para tratar como o sistema deve ser modelado, e psicologia, sociologia e antropologia, com o propósito de compreender o usuário final, suas limitações e tendências.

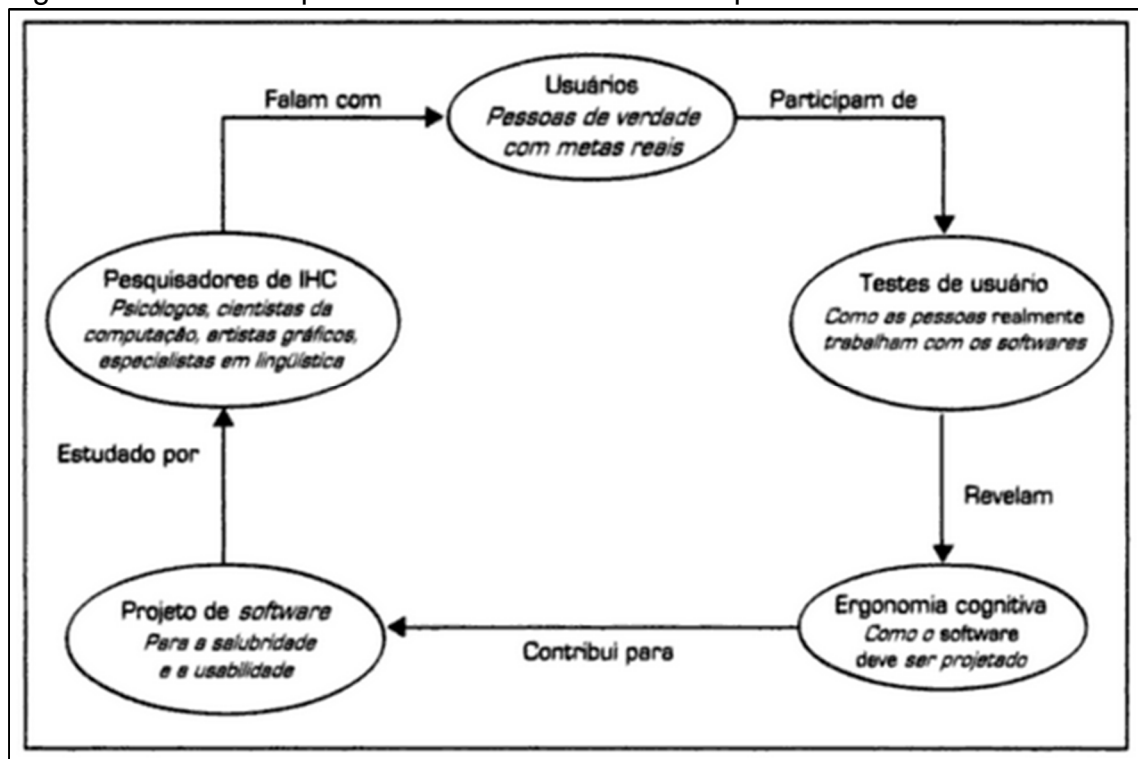
Os autores ainda retratam os benefícios da IHC, citando a possibilidade de aumentar a produtividade do usuário ao interagir com um sistema bem projetado, aproveitar as características humanas e utilizá-las a favor do desenvolvimento, satisfazer suas necessidades e aumentar, de forma geral, o bem-estar e a experiência pessoal do usuário.

[...] em muitos casos, o computador, ao forçar os usuários a aprender a língua dele, exerce uma forma de poder sobre os indivíduos estabelecendo uma unilateralidade de adaptação. Em outras palavras, as pessoas devem adaptar-se aos computador, resultando em uma interação e em um aprendizado artificial, insalubre e improdutivo. (BAWA, 1997, p. 42).

A justificativa de Bawa (1997) sobre a importância do estudo de IHC evidencia seu ponto de vista sobre o assunto. Partindo de sua perspectiva e com base em sua citação, entende-se que existe um problema elementar na interação primitiva entre o homem e a máquina. Isso se dá pelo fato de que o ser humano cresce e evolui em seu tempo de vida, e é esperado que a tecnologia, de alguma forma, faça o mesmo.

Como revelado anteriormente, a disciplina de IHC conta com indivíduos, incluindo profissionais em diversas áreas e usuários comuns que fazem o uso do sistema. Assim, o desenvolvimento de uma boa *interface* não é resultado de um bom desenvolvedor de sistemas, mas sim do esforço coletivo de diversos indivíduos que podem contribuir com o produto final. Este ciclo pode ser observado na Figura 7 que demonstra a forma como uma interface pode ser refinada.

Figura 7 – Ciclo de aprimoramento de um software partindo de uma boa IHC.



Fonte: BAWA (1997, p. 43.).

4.7 TESTE DE SOFTWARE (TS)

A etapa de teste de *software* é caracterizada pela revisão, de certo modo, de algum dos aspectos que foram desenvolvidos anteriormente. É essencial para a engenharia de *software* que o TS seja regulamentado de alguma forma com o objetivo de padronizar o controle de qualidade que é feito sobre o produto.

Por melhores que sejam as técnicas de modelagem e especificação de *software*, por mais disciplinada e experiente que seja a equipe de desenvolvimento, sempre haverá um fator que faz que o teste de *software* seja necessário: *o erro humano*. É um mito pensar que bons desenvolvedores, bem concentrados e com boas ferramentas serão capazes de desenvolver *software* sem erros. (Beizer, 1990, citado por Wazlawick, 2013, p. 289).

Tendo o ponto de vista do autor sobre o *software* como um produto fadado à imperfeição em seu lançamento, é compreensível que sejam estabelecidas métricas para controlar a qualidade do *software* como produto final.

Pezzè e Young (2008) deixam claro que o período de testes é um processo repetitivo que deve acontecer múltiplas vezes com o propósito de validar a funcionalidade, disponibilidade, usabilidade e segurança de um *software*. Também é preciso manter em mente que o TS ocorre também quando há atualizações em um sistema já consolidado, reavaliando o código-fonte novo que foi implementado e também o antigo, com o propósito de garantir que não houve nenhum tipo de conflito entre as duas partes.

O teste proporciona o último elemento a partir do qual a qualidade pode ser estimada e, mais pragmaticamente, os erros podem ser descobertos. Mas o teste não deve ser visto como uma rede de segurança. [...] A qualidade é incorporada ao *software* através do processo de engenharia de *software*. (Pressman, 2011, p. 402).

Pressman (2011), em sua citação, se refere ao fato de que o TS não gera qualidade em um *software*, mas sim uma avaliação da situação na qual o *software* se encontra, para que então, partindo de tal perspectiva, seja possível aprimorá-lo através do *feedback*.

4.7.1 Tipos de teste de software

A Engenharia de *Software* apresenta uma gama extensa de testes de *software* que podem ser aplicados para um determinado sistema. Wazlawick (2013) trata dos mais populares no âmbito do desenvolvimento:

- a) **Teste de Unidade:** este tipo de teste é considerado básico e geralmente é realizado pelo próprio programador, pois, segundo o autor, consiste em verificar a funcionalidade de componentes individuais de um *software* – entende-se um componente de um *software* como sendo um pacote, classe ou método;
- b) **Teste de Integração:** este teste tem como propósito verificar o estado da comunicação entre múltiplos componentes de um *software*, averiguando também se há conflitos entre as partes que estão sofrendo a união;
- c) **Teste de Sistema:** de acordo com o autor, trata-se do tipo de teste que verifica a disponibilidade em executar as operações solicitadas por um usuário em um determinado sistema;
- d) **Teste de Aceitação:** em oposição aos demais, este teste é realizado pelo usuário final do sistema, pois tem como propósito aprovar o produto que esteve em desenvolvimento;
- e) **Teste de Performance:** este tipo de teste, como o nome já diz, tem como propósito avaliar o desempenho do sistema. De acordo com o autor, mesmo que não haja requisitos de desempenho para serem utilizados como parâmetro, é importante que sejam realizadas tentativas de sobrecarga e estresse com o objetivo de testar o tempo de resposta nesse contexto;
- f) **Teste de Segurança:** Wazlawick (2013) considera o teste de segurança como aquele que abrange toda parte de segurança que um sistema deve possuir, embora o divida em seis aspectos: integridade, autenticação, autorização, confidencialidade, disponibilidade e não repúdio;
- g) **Teste Funcional:** O teste funcional não exerce qualquer avaliação sobre a estrutura do sistema, mas sim sobre a funcionalidade do mesmo em função do que foi estabelecido no contrato inicial entre cliente e desenvolvedor. O autor

destaca que este teste é um dos pilares de integridade que validam a engenharia de *software* como prática que garante a viabilização do mercado de *software*.

Observa-se a partir dessa listagem que existem diversas opções para se realizar a validação de um *software*. No entanto, é importante observar que os testes citados não substituem um ao outro. Embora um determinado contexto de desenvolvimento não necessariamente requeira a aplicação de todos os tipos de teste de *software* disponíveis, a aplicação de um teste não isenta o profissional de realizar qualquer outro teste, tendo em vista o fato de que eles tratam de aspectos diferentes do sistema.

5 ALGORITMO

Algoritmos são elementos essenciais à computação como a entendemos nos dias de hoje e, graças a eles, podem ser tecidos raciocínios lógicos completos, além de estabelecer rotinas e expressar a forma como um processador deve reagir para cada caso específico de situação prevista. Brookshear (2013, p. 2) define informalmente um algoritmo como “um conjunto de passos com os quais uma tarefa pode ser realizada.”.

No entanto, a palavra algoritmo em si não surgiu no contexto computacional. Seu surgimento pode ser datado antes mesmo da computação como a conhecemos nos dias de hoje.

“[...] a palavra algoritmo vem do nome do matemático iraniano Abu Abdullah Mohammad Ibn Musa al-Khawarizmi, nascido em Khawarizm (Kheva), ao sul do mar Aral, que viveu no século XVII, o que nos leva a avaliar o fato de que conjuntos de ações denotativas e objetivas precedem a existência das máquinas de calcular.” (MEDINA; FERTIG, 2005, p. 13).

No contexto em que o termo será tratado, Manzano e Oliveira (2005) definem algoritmo de forma sucinta como um processo de cálculo matemático ou da descrição sistemática da resolução de um grupo de problemas semelhantes, e complementam a definição afirmando que um algoritmo também pode ser visto como regras formais para obtenção de um resultado ou da solução de um problema englobando fórmulas de expressões aritméticas.

Assim sendo, pode-se concluir que, no respectivo contexto, de uma forma geral, algoritmos são sequenciais de regras e/ou instruções claras e bem definidas que existem com o propósito final de executar determinada rotina.

Gunnerson (2001) compara linguagens de programação com ferramentas de trabalho; a partir de seu ponto de vista, cada ferramenta desempenha uma função específica e, embora exista a possibilidade de utilizar uma determinada linguagem em um contexto em que ela não foi projetada para ser aplicada, é mais eficiente que se faça uso da linguagem apropriada partindo de ambos os pontos de vista de compatibilidade e ferramentas disponíveis partindo de tal linguagem.

5.1 O PARADIGMA DA ORIENTAÇÃO A OBJETOS

O avanço na tecnologia com a qual *softwares* são produzidos acabou por implementar novos conjuntos de regras, costumes, hábitos e práticas que substituem as antigas, possibilitando assim que existam continuamente produtos melhores que surgem a partir de técnicas melhores. O paradigma da orientação a objetos é um exemplo claro deste caso.

“Um paradigma é um conjunto de regras que estabelecem fronteiras e descrevem como resolver problemas dentro desta fronteira. Um paradigma ajuda-nos a organizar e a coordenar a maneira como olhamos o mundo.” (SOUZA, 2016, p. 7).

Considerando este conceito, é possível compreender o que é a orientação a objetos. De acordo com Souza (2016), trata-se de um paradigma utilizado no desenvolvimento de *software* baseado em componentes individuais com o propósito de criar sistemas mais complexos.

Dentro do contexto de programação de computadores, uma linguagem de programação funciona como um meio de comunicação entre o indivíduo e um determinado modelo de computador, fazendo a ligação entre o pensamento humano e o modelo requerido para o processamento pela máquina (TOSCANI, 2001, citado por CONTI, 2014).

Segundo Gudwin (1997), programação orientada a objeto é uma técnica de programação que surgiu devido à necessidade de se organizar o processo de programação, sendo que esta alterou o padrão de engenharia de *software* vigente na época, que originalmente derivavam os módulos de acordo com a funcionalidade do sistema, sendo principalmente módulos procedimentais que sendo alimentados com dados geravam novos dados, porém após o advento da orientação a objetos o conceito de objeto foi introduzido na engenharia de *software*. (CONTI, 2014).

Segundo Lafore (2002), a programação orientada a objetos oferece uma nova e poderosa forma de lidar com complexidade. Ao invés de ver um programa como uma série de instruções a serem executadas, ele é visto como um grupo de objetos que tem determinadas propriedades e pode realizar determinadas ações, o que resulta em programas mais limpos, confiáveis e fáceis de serem mantidos.

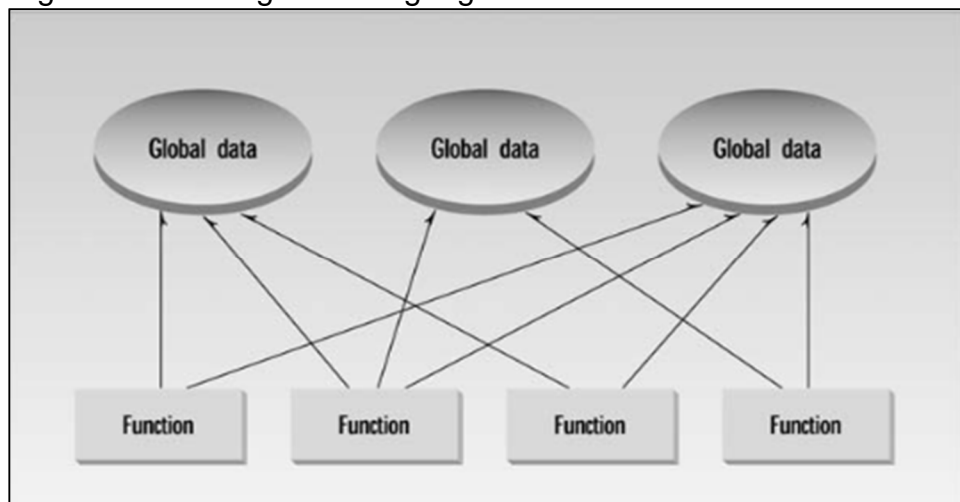
Programação orientada a objetos (POO) é um paradigma de programação que permite os programadores raciocinar e solucionar problemas em termos de objetos, os quais estão diretamente associados às entidades ou 'coisas' reais. E, como resultado desse mapeamento natural, o programador pode se concentrar mais na solução do problema em vez de tentar vislumbrar o sistema como um conjunto de funções e dados. (MENDES, 2010).

A programação orientada a objetos foi necessária a partir do momento em que a tecnologia avançou e deu espaço para o surgimento de programas maiores e mais complexos, pois gradativamente a forma tradicional de se programar – lê-se o paradigma de programação estruturada – na qual cada linha diz ao *software* um comando para ser executado, os profissionais da época começaram a se deparar com limitações e problemas de logística.

5.1.1 Embasamento para o uso do paradigma atual

Lafore (2002) afirma que existem duas falhas principais no paradigma de programação estruturada. O primeiro deles é o fato de que as funções tinham acesso livre e completo aos dados de variáveis globais da execução do *software*. O paradigma da linguagem estruturada pode ser observado na Figura 8.

Figura 8 – Paradigma de linguagem estruturada.



Fonte: Lafore (2002, p. 11).

Esta questão é um problema por tornar o programa extremamente difícil de ser conceitualizado e de ser modificado, dado ao fato de que qualquer alteração em qualquer ponto do programa pode resultar em diferentes comportamentos em outros aspectos dele.

A segunda grande falha da programação estruturada é que seu arranjo de informações não expressa de forma eficiente a organização de dados do mundo real, pois não trata de atributos (variáveis locais de um objeto) e comportamento (métodos).

Cada objeto modela algum aspecto do programa que você está tentando resolver. Escrever listas sequenciais de chamadas de procedimento para dirigir o fluxo do programa não é mais o foco da programação sob a orientação a objetos. Em vez disso, os objetos interagem entre si, para orientar o fluxo global do programa. (SINTES, 2010, p. 6).

5.2 A PLATAFORMA .NET

A plataforma .NET é um *framework* que permite a execução e compatibilidade de programas para diversos sistemas operacionais, aceitando a interação entre diferentes linguagens de programação através de sua estrutura.

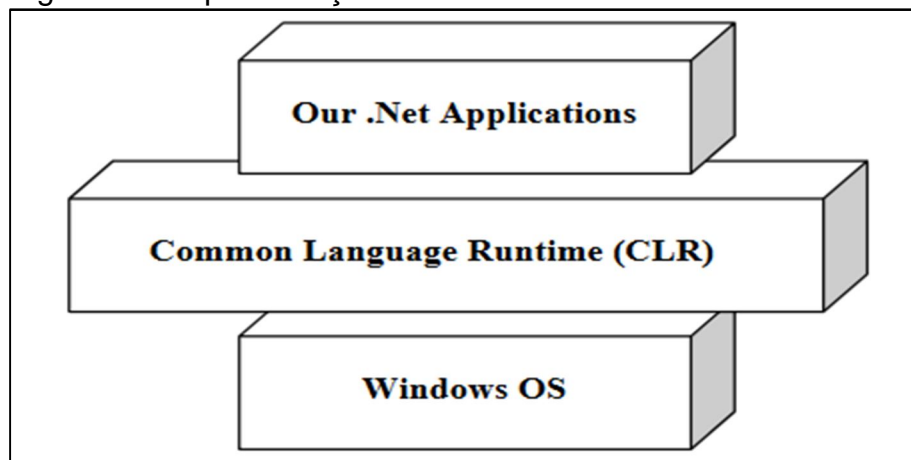
O .NET Framework é um ambiente de execução gerenciado que fornece uma variedade de serviços para os aplicativos em execução. Ele consiste em dois componentes principais: o common language runtime (CLR), que é o mecanismo de execução que gerencia a execução de aplicativos; e a biblioteca de classes do .NET Framework, que fornece uma biblioteca de testados, código reutilizável que os desenvolvedores possam chamar de seus próprios aplicativos. (MSDN, portal¹).

De acordo com Liberty (2005), o *Framework* .NET é um grande avanço da tecnologia por não apenas suportar linguagens de programação independentes, como também a integração entre elas, o que significa que é possível herdar classes, capturar exceções e utilizar polimorfismo através de linguagens diferentes.

¹ Disponível em: <<https://msdn.microsoft.com/pt-br/library/hh425099%28v=vs.110%29.aspx>>.

Segundo o portal MSDN², o *.NET Framework* tem como alicerce a *Common Language Runtime*, responsável por executar todo código gerenciável (passível de interpretação em tempo de execução).

Figura 9 – Representação da CLR.

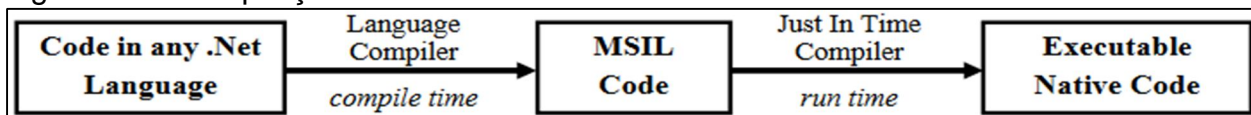


Fonte: Rasheed (2006, p. 18).

Como pode ser observado na Figura 9, a CLR tem como papel fundamental servir de mediador para a execução do código-fonte, convertendo-o para uma linguagem intermediária Microsoft Intermediate Language (MSIL), que é interpretada pela Common Language Runtime para então ser convertida em código binário executável.

Rasheed (2006) explica o processo de execução de um trecho de código em um ambiente .NET em conformidade com o esquema apresentado na Figura 10.

Figura 10 – Compilação no ambiente .NET.



Fonte: Rasheed (2006, p. 18).

Levando em conta o esquema da Figura 10 e comparando-o com a representação da Figura 9, é possível estabelecer a relação entre os componentes e as

² Disponível em: <<https://msdn.microsoft.com/pt-br/library/zw4w595w%28v=vs.110%29.aspx>>.

suas respectivas etapas. As aplicações .NET geram a linguagem comportada pelo *Framework .NET*, a *Common Language Runtime* interpreta o código de linguagem intermediária, e o sistema operacional acaba ficando responsável pelo código binário executável.

5.2.1 A Linguagem de programação C#

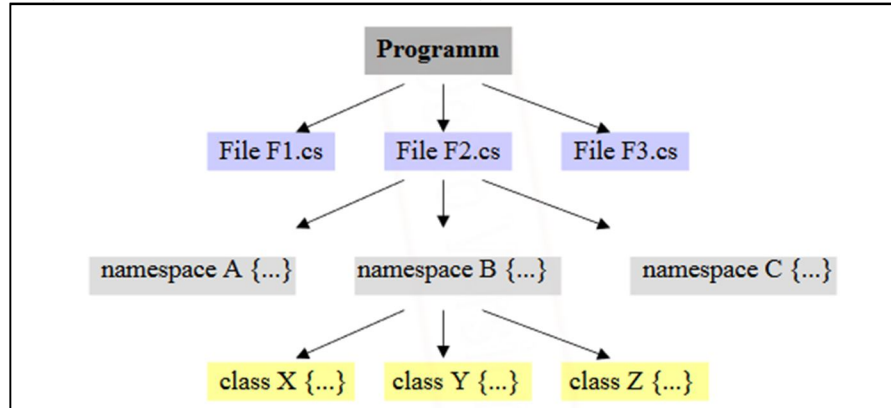
Sharp (2010) introduz em sua obra o C# como uma linguagem simples e poderosa, que tem como público alvo desenvolvedores criando aplicações fazendo uso do Microsoft NET *Framework*. Ele complementa dizendo que essa linguagem herda as melhores características do C++ e do Microsoft Visual Basic, com a vantagem de ter menos inconsistências e anacronismos, o que acaba gerando uma linguagem mais clara e lógica.

Sobre C#, Rasheed (2006) diz que C# é, sem dúvidas, a melhor escolha de linguagem para o ambiente .NET, e complementa afirmando que ela combina o poder e eficiência do C++, a simplicidade e clareza dos conceitos de *design* de orientação a objeto do Java e a simplificação de linguagem do Visual Basic.

Tendo em mente essa definição, é possível compreender o porquê de muitos autores e especialistas no assunto concordarem sobre o fato de que a linguagem C# é extremamente flexível dentro de seu ambiente, sendo capaz até mesmo de reutilizar códigos escritos em Visual Basic.NET, C++ gerenciado ou F#, pois todas essas linguagens compartilham diversos aspectos que possibilitam essa interação.

Como pode ser visto na Figura 11, um programa desenvolvido em C# consiste na estrutura demonstrada. Cada arquivo de extensão “.cs” possui uma determinada quantidade de “áreas de trabalho” (*namespaces*) nos quais uma determinada quantidade de classes são reconhecidas.

Figura 11 – Estrutura de um programa em C#.



Fonte: Mössenböck (2001, p. 6).

5.3 A LINGUAGEM PHP

O PHP é uma linguagem de criação de *scripts* código aberto, que é flexível ao ponto de se adequar a diversos propósitos, desde a criação de páginas web até mesmo a chamada de funções para aplicações móveis.

De acordo com Dall'Oglio (2015), embora a linguagem tenha sido criada com um propósito específico de gerenciar uma página na internet, a linguagem nunca parou de crescer e passou por diversas reescritas tendo a finalidade de se adaptar ao desenvolvimento da tecnologia, o que permitiu que ela sobrevivesse ao tempo.

Niederauer (2011) considera o PHP como uma das linguagens mais utilizadas no contexto web, complementando o HTML em páginas que requerem funções mais complexas, além de resolver problemas logísticos que seriam inviáveis para o HTML implementado sozinho.

O autor ainda ressalta a vantagem de que o código-fonte e a documentação do PHP estão disponíveis no próprio *site* do desenvolvedor gratuitamente, o que viabiliza a customização da linguagem.

Niederauer (2011) faz menção aos diferentes contextos em que a linguagem PHP pode ser implementada, sendo as principais: páginas *web*, servidores de aplicação e *scripts* para manipulação de bancos de dados.

6 O BANCO DE DADOS ESTRUTURADO

Heuser (2009, p. 22), a respeito da definição de banco de dados, afirma que se trata do “[...] conjunto de dados integrados que tem por objetivo atender a uma comunidade de usuários.”.

Essa afirmação dá a entender que um banco de dados deve estar aberto e disponível para que outros usuários que solicitem essa informação tenham pleno acesso a elas, pois isso é necessário com o propósito de evitar a redundância de informações e a inconsistência dos dados armazenados.

[...] pode ser considerado como o equivalente eletrônico de um armário de arquivamento; ou seja, ele é um repositório ou recipiente para uma coleção de arquivos de dados computadorizados. Os usuários de um sistema podem realizar (ou melhor, solicitar que o sistema realize) diversas operações envolvendo tais arquivos. (DATE, 2004, p. 3).

O autor estabelece uma metáfora que permite uma melhor compreensão sobre bancos de dados. Tal metáfora ajuda na compreensão sobre como funciona, na prática, a interação com um banco de dados.

Elmasri e Navathe (2012) expressam com êxito a importância dos bancos de dados na vida de qualquer pessoa quando afirmam que este recurso computacional está presente de forma crítica em praticamente qualquer área da computação.

Os autores, ao afirmarem tal posição, se referem ao fato de que toda forma de tecnologia requer o armazenamento e manipulação de dados e, para que isso seja concretizado, a forma mais viável até os dias de hoje é a utilização de um banco de dados.

Oliveira (2002) reforça esse ponto de vista definindo que para a definição de qualquer sistema, é esperado que haja entrada e saída de dados processados e que, com o propósito de se adequar às necessidades geradas pelo avanço da tecnologia, junto às expectativas do usuário em função desses avanços, passou a ser fundamental o armazenamento e manipulação de informações.

Sobre a forma como os bancos de dados são convencionalmente estruturados, Batimarchi (2015) define uma linha de raciocínio que pode ser compreendida. De

acordo com o autor, trata-se do conjunto de dados cujo existe uma organização para que sejam recuperados, tal como etiquetas, linhas e colunas que possibilitam ao administrador de banco de dados a simplificação de seu trabalho.

6.1 O BANCO DE DADOS RELACIONAL

O propósito por trás da existência de bancos de dados relacionais está no fato de que em muitos casos diferentes tabelas estão de alguma forma relacionadas, seja no contexto de tabelas “pai e filhas”, no qual existem diversas entradas de dados para um único dado da tabela pai, ou no contexto de buscas e relacionamentos indiretos que podem ser criados a partir de dada necessidade.

O conceito de bancos de dados relacionais está na forma em que eles são implementados, que estabelece uma relação lógica entre os dados, para que a repetição de dados (redundância) seja a menor possível, economizando espaço em disco e aumentando a velocidade de consulta dos dados. (PORTAL EDUCAÇÃO³, 2013).

De acordo com o colunista, este modelo de banco de dados possui como entidade central as próprias tabelas que o compõe. Desta forma, as colunas ficam responsáveis por determinar o tipo de dado que está sendo armazenado e as linhas (ou tuplas, como são chamadas) representam cada novo dado que é inserido naquela determinada tabela.

Segundo Date (2004, p. 7), “Um banco de dados relacional é simplesmente um banco de dados que é percebido pelo usuário como um grupo de tabelas – onde uma tabela é um grupo não ordenado de linhas”.

Trata-se, portanto, de acordo com a definição de Date (2004), do sistema organizacional tradicional conhecido no âmbito da ciência da computação, e do desenvolvimento de sistemas em geral no qual existe um universo chamado Banco de dados, e as tabelas de dados se relacionam dentro desse universo conforme há necessidade de tecer conexões entre eles.

³ Disponível em: <<http://www.portaleducacao.com.br/informatica/artigos/46246/bancos-de-dados-relacionais>>.

6.2 A LINGUAGEM SQL

Embora as primeiras versões da linguagem SQL não incluam as práticas envolvidas no padrão de bancos de dados relacionais, Elmasri e Navathe (2012) afirmam que ela ainda pode ser considerada como um dos principais fatores que levaram ao sucesso dos bancos de dados relacionais comerciais, dado ao fato de que a linguagem acabou se tornando um padrão para a estruturação dos sistemas de gerenciamento de banco de dados, o que de certa forma dá mais flexibilidade ao cliente quando resolve escolher por um específico sistema.

De acordo com Oliveira (2002), a linguagem SQL é um conjunto de comandos – expressados através de uma semântica e respeitando uma determinada sintaxe – que tem como objetivo manipular bancos de dados. O autor ainda ressalta que essa linguagem poderia ser considerada por muitos profissionais da área como uma sublinguagem, pois quando aplicada em um contexto, ela deve sempre acompanhar uma linguagem de programação. Neste caso, a SQL é utilizada em um programa quando há interação com o banco de dados.

Entende-se, neste caso, que a interação com um banco de dados pode se manifestar nas formas de manter e criar uma estrutura – tabela de dados ou banco de dados, além de manipular os dados através de variações das propriedades básicas de inclusão, exclusão, modificação e busca.

6.3 SISTEMAS DE GERENCIAMENTO DE BANCOS DE DADOS (SGBD)

Com a ascensão da ideia de se armazenar e manipular os dados que são tratados pelos grandes sistemas de processamento de dados através da aplicação dos conceitos de banco de dados, surgiram os sistemas voltados ao seu gerenciamento, cujo um dos principais propósitos é de garantir a integridade das informações que são tratadas.

[...] um sistema de banco de dados é basicamente um sistema computadorizado de manutenção de registros; em outras palavras, é um

sistema computadorizado cuja finalidade geral é armazenar informações e permitir que os usuários busquem e atualizem essas informações quando as solicitar. (DATE, 2004, p. 6).

Heuser (2009, p. 23) define um sistema de gerência de banco de dados como um “[...] *software* que incorpora as funções de definição, recuperação e alteração de dados em um banco de dados”.

Essa definição, apesar de ser generalista basta para que seja compreendida a importância desses sistemas. Trata-se do componente auxiliar que serve como ponte de comunicação entre o sistema de informação e o banco de dados, desempenhando no banco todas as funções que são pedidas pelo sistema e retornando um ou mais dados quando necessário.

Elmasri e Navathe (2012, p. 3) definem sistemas gerenciadores de bancos de dados como “[...] uma coleção de programas que permite aos usuários criar e manter um banco de dados.”.

Ambas as definições dos autores mencionados remetem ao fato de que um SGBD é flexível e tem como propósito a servir de ferramenta para administrar os dados de um sistema. Podemos entender “manter” um banco de dados como as interações de inserir, excluir, alterar e buscar dados em tabelas.

Além de ser essencial para a manipulação de um banco de dados já estabelecido, um SGBD ainda pode ter outros benefícios, como o controle de redundâncias e políticas de segurança que garantem a integridade da informação armazenada e o controle de acesso a essa informação.

6.4 O BANCO DE DADOS MYSQL

Uma das ferramentas mais conhecidas e utilizadas pelos profissionais de informação é o banco de dados MySQL. Esta ferramenta com relativo baixo custo de implementação e manutenção é visto por esses profissionais como sendo uma ferramenta de alta confiabilidade e estabilidade, além de ser versátil e ter compatibilidade com grande parte dos sistemas que existem hoje.

De acordo com Suehring (2002), o MySQL apresenta “o melhor de todos os mundos” quando se trata de sua perspectiva em função das outras opções disponíveis no mercado. Isso se dá ao fato de que ele é plenamente estável, possui baixo custo de implementação e desempenha seu papel em múltiplas plataformas. O autor ainda complementa com o fato de que sua documentação é excepcional, assim como o suporte ao programa.

Essas características destacadas pelo autor são levadas em consideração como sendo de extrema importância quando é cogitado o uso de um sistema de gerenciamento de banco de dados em um determinado contexto. Isso se deve ao fato de que são aspectos que influenciam na implementação de um sistema de informação e/ou em seu desempenho.

Widenius e Axmark (2002) chamam a atenção para o fato de que o MySQL é uma iniciativa de código aberto, o que pode permitir uma otimização ainda mais além das expectativas do usuário, dada possibilidade de extrair maior valor sobre seus recursos.

A ideia dos autores em criar a variação MySQL surgiu a partir da utilização do sistema de gerenciamento mSQL. No entanto, segundo os autores, após análise foi concluído que esse sistema não era rápido e flexível o bastante para preencher suas necessidades.

7 INTELIGÊNCIA ARTIFICIAL (IA)

Segundo Rosa (2011), o desafio em definir Inteligência Artificial está na palavra “inteligência”, dado ao fato de que se pode considerar “artificial”, de forma simplificada e generalista, aquilo que é criado pelo homem. Desta forma, o autor cita Rich (1994) em seu dizer de que “IA é o estudo de como fazer os computadores realizarem tarefas as quais, até o momento, os homens fazem melhor”.

De acordo com este raciocínio, pode-se concluir que de maneira geral a Inteligência Artificial é a área de estudo que se destaca na análise e desenvolvimento de métodos computacionais que possibilitam à máquina solucionar problemas fazendo o uso do ponto de vista humano.

Ertel (2011) avalia a citação de Rich (1994) dizendo que seria perigoso concluir que a definição do segundo autor se refere apenas à implementação pragmática de processos inteligentes. No sentido de sua definição, Ertel (2011) afirma que um sistema inteligente não pode ser construído sem uma compreensão mais profunda sobre o raciocínio humano, e este ponto é o que torna o estudo da neurociência um grande fator para a Inteligência Artificial.

Carvalho, A. C. P. L. F. et al. (2011) consideram que o potencial de aprendizado é um fator essencial para que um indivíduo seja considerado inteligente. Memorizar, observar, evoluir e reorganizar o conhecimento são aspectos que fazem parte desse potencial de aprendizado.

Embora o conceito de inteligência seja nebuloso e extremamente polêmico, pode-se afirmar que comportamento inteligente é aprender a lidar com o mundo, é desenvolver estratégias para encontrar soluções, é a capacidade para lidar com novidades. A inteligência não é atributo específico do ser humano, embora este o tenha em mais alto grau. (SANVITO, 1995, p. 362).

A perspectiva de Sanvito (1995) não coloca o ser humano como indivíduo exclusivo no que se trata de inteligência. Sua afirmação, embora simples, é alvo de discussões até mesmo nos dias de hoje sobre a relação entre a inteligência que se compreende nos dias de hoje e a que pode ser desenvolvida, por exemplo, em laboratórios no caso da Inteligência Artificial.

7.1 A RELAÇÃO COM A INTELIGÊNCIA HUMANA

Faria (2003) afirma que não existe uma concordância plena entre os estudiosos da área – psicólogos e educadores – sobre os tipos de inteligência que podem se manifestar em um ser humano.

No entanto, é evidente a relação entre essa inteligência humana e a lógica. Sanvito (1995), a respeito da lógica, afirma que a busca pela racionalidade é um aspecto constante do pensamento humano, e que é comum um indivíduo descartar aquilo que não é considerado logicamente correto. No entanto, o autor questiona o cérebro humano como “metabolizador de informações”, cujo funcionamento é essencialmente dependente de princípios lógicos e matemáticos.

O autor complementa seu ponto de vista evidenciando o fato de que a mente humana é capaz de lidar com a ambiguidade, a incerteza, a desordem e o incompleto e que, portanto, não se relaciona exclusivamente com os aspectos lógicos do meio ao seu redor. Essa capacidade adaptativa é dita como um dos mecanismos de evolução da mente. Em um meio em que há exclusivamente um conjunto pré-determinado de regras lógicas e matemáticas, no entanto, não é possível que exista evolução, pois a mente fica limitada ao que é estabelecido.

Partindo dos conceitos estabelecidos e das definições clássicas sobre Inteligência Artificial, é possível estabelecer a relação entre as partes. A IA possibilita, através de métodos diversos que simulam o comportamento da mente humana, uma conclusão exclusivamente lógica baseada em um algoritmo estabelecido a partir de uma determinada entrada de dados. A inteligência humana, por outro lado, permite a resolução de problemas através dos mecanismos evolutivos de dedução e aprendizado natural.

7.2 HISTÓRIA DA INTELIGÊNCIA ARTIFICIAL

Bittencourt (2006), a respeito do histórico da disciplina de IA, faz um paralelo com estudos humanos muito mais antigos – no caso, a filosofia, a matemática e a tecnologia – para explicar que existe uma relação semelhante com a qual essas três

áreas se relacionam (sendo a tecnologia fruto da filosofia em junção da matemática). Assim, a Inteligência Artificial é composta pela junção do desenvolvimento de modelos cognitivos e a implementação de aplicações.

A relação imposta pelo autor é pertinente por demonstrar a forma como a área de Inteligência Artificial surge no âmbito da computação de forma prática. Ainda que a origem da disciplina seja muito diferente do que é dito – em um tempo em que a palavra “aplicação” não poderia ser compreendida da forma como é compreendida atualmente, trata-se de um ponto que mostra a forma como a IA (ainda como conceito absolutamente teórico) passou a ser unida à tecnologia.

Sobre o desenvolvimento da área de IA aplicada, Gongora (2007) data o início junto ao desenvolvimento das tecnologias de guerra no decorrer da segunda guerra mundial. Durante o período, criou-se a necessidade de máquinas de cálculo que realizassem o processamento da trajetória de mísseis, bem como, de máquinas que efetuassem simulações e auxiliassem o planejamento estratégico. Essa necessidade abriu portas para a aplicação de uma Inteligência Artificial extremamente primitiva, a partir da formalização matemática de um neurônio (lê-se o neurônio formal).

Chrisley (2000) complementa essa ideia afirmando que o nascimento da Inteligência Artificial aconteceu entre os períodos de 1910 – 1945, quando o embate entre a lógica e a psicologia estava em alta. Neste período, houve uma separação entre o concreto e o raciocínio e tal discussão contribuiu diretamente para o desenvolvimento da IA.

É interessante observar que a visão de Gongora (2007) e de Chrisley (2000) sobre o assunto segue o paralelo traçado por Bittencourt (2006), quando o autor diz que a IA é produto da filosofia e da matemática aplicados à tecnologia. Analisando o contexto apresentado, é possível identificar essas entidades. A filosofia e a matemática são representadas pelo embate da lógica em oposição à psicologia, enquanto a tecnologia pode ser identificada como as máquinas de guerra que abriram a visão dos cientistas sobre as possibilidades eminentes a respeito da IA.

7.3 APLICAÇÕES DA INTELIGÊNCIA ARTIFICIAL

Considerando a definição sobre a Inteligência Artificial e suas possíveis aplicações, torna-se perceptível a abrangência com a qual a disciplina pode ser aplicada. Se é possível afirmar que a tecnologia tem a capacidade de tornar qualquer área de estudo ou de atuação profissional otimizada, isso quer dizer que a IA também tem essa mesma capacidade, pois se trata de um aprimoramento em função da informatização.

Nilsson (1980), partindo do ponto de vista acadêmico, lista algumas das aplicações da disciplina de Inteligência Artificial, sendo elas:

- a) Extração de dados inteligente;
- b) Processamento de linguagem natural
- c) Sistemas especialistas;
- d) Demonstração de teoremas;
- e) Robótica;
- f) Programação automática;
- g) Problemas de análise combinatória;
- h) Problemas de percepção.

O autor levanta os tópicos que, atualmente, são consolidados como áreas de estudo da Inteligência Artificial, ou ainda aplicados em outras áreas como a tecnologia administrativa e a matemática.

Martins (2010) afirma em relação à Inteligência Artificial sendo aplicada em tecnologias administrativas, que o estudo de sistemas oferece um grande potencial para o crescimento da ciência da informação. Isso se dá ao fato de que permite ao usuário comum ter acesso a mais conteúdo relevante através de um sistema que é treinado para filtrar e disponibilizar informações de qualidade.

A Inteligência Artificial também pode ser empregada em sistemas que permitem a classificação automática de conteúdos, uma vez que para serem recuperados os conteúdos precisam de uma organização que seja tanto lógica quanto semanticamente coerente. No mundo digital as Ontologias cobrem esse papel e podem ser criadas tanto de forma manual quanto automática. (MARTINS, 2010, p. 13).

7.4 PROCESSAMENTO DE LINGUAS NATURAIS (PLN)

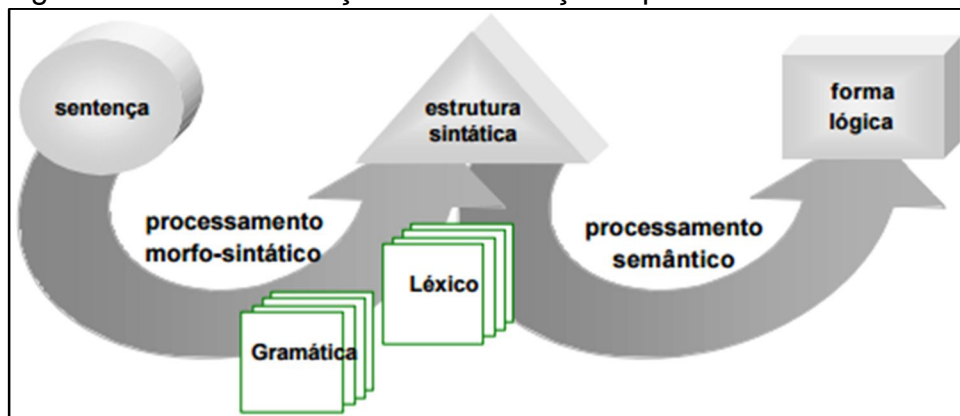
O interesse do meio acadêmico sobre o estudo do PLN se dá pelo fato de que essa área de estudo permite uma aproximação que antes não foi vista da linguagem casual utilizada no cotidiano.

O processamento da linguagem natural (PLN) trata computacionalmente os diversos aspectos comunicação humana, como sons, palavras, sentenças e discursos, considerando formatos e referências, estruturas e significados, contextos e usos. Em sentido bem amplo, podemos dizer que o PLN visa fazer o computador se comunicar em linguagem humana, nem sempre necessariamente em todos os níveis de entendimento e/ou geração de sons, palavras, sentenças e discursos. (GONZALEZ; LIMA, 2003).

A descrição proposta por Gonzalez e Lima (2003) pode ser vista justamente pela perspectiva de aproximação e da compreensão da língua natural por parte de um indivíduo computacional. Os autores ainda mencionam os níveis de entendimento da linguagem natural, sendo eles: fonético, morfológico, sintático, semântico e pragmático.

A Figura 12 apresenta a forma como Gonzalez e Lima (2003) apresentam o PLN, desde a captura de uma sentença até o significado lógico do que foi dito.

Figura 12 – Transformações da sentença no processamento.



Fonte: GONZALEZ; LIMA (2003).

Segundo Rosa (2011), o processamento de linguas naturais pode ser definido de várias formas, embora todas elas, de certo modo, convirjam para um mesmo propósito:

treinar um sistema computacional para que tenha a possibilidade de processar a linguagem que é utilizada pelos humanos no cotidiano.

De acordo com o autor, é impossível definir PLN sem que se leve em conta às noções de armazenamento de dados e de manipulação linguística desses dados. Pode-se entender esses dois aspectos, portanto, como primordiais para que seja estabelecido um sistema que processe línguas naturais.

Lima, Nunes e Vieira (2007) fazem uma relação direta entre a popularização de aplicações que lidam com informação ao processamento de línguas naturais. De acordo com os autores, essa é uma das soluções viáveis para lidar com grandes volumes de informação.

Partindo desse ponto de vista, entende-se o PLN como um meio para diversos tipos de aplicação tendo como base a possibilidade de criar sistemas que compreendem a estrutura linguística utilizada pelos usuários.

De acordo com Silva (2014), existem quatro aspectos básicos que juntos formam um analisador de linguagem natural capaz de compreender sentenças em um computador, sendo elas: análise léxico-morfológica, sintática, semântica e pragmática. Essas quatro são vistas em detalhes a seguir:

- a) **Análise léxico-morfológica:** o analisador léxico-morfológico divide a sentença em itens lexicais e realiza uma varredura, buscando tratar item a item, e decompondo-os em seus morfemas. Verifica ainda, através das informações contidas no léxico e nos morfemas, a estrutura, características e informações associadas a um determinado item, tais como gênero e número para substantivos, ou pessoa, número, modo e tempo, para os verbos, por exemplo. (LIMA; VIEIRA, 2008?, citado por SILVA, 2014).
- b) **Análise sintática:** na análise sintática, sequências lineares de palavras são transformadas em estruturas que ditam como as palavras se relacionam entre si. Ao violar as normas da linguagem que regem a combinação de palavras, algumas sequências de palavras poderão ser rejeitadas. (RICH, 1988, citado por SILVA, 2014).
- c) **Análise semântica:** Vaz (2004) relata que o analisador semântico é responsável por encontrar o sentido da estrutura de palavras reagrupadas pelo analisador

sintático, uma vez que as palavras já foram identificadas individualmente. Porém, o ponto onde a interpretação semântica será produzida é um problema a ser resolvido. Uma alternativa é gerar uma interpretação sintática completa e depois passar esta estrutura para um interpretador semântico. (SILVA, 2014).

- d) **Análise pragmática:** Esta etapa é a última e a mais difícil a se solucionar. A Análise Pragmática além de verificar toda a parte gramatical e o sentido da frase, ela também verifica a sua coerência. Uma frase pode satisfazer as etapas anteriores, porém, uma frase como “A Alemanha é tricampeã da Copa América” é totalmente fora de nexos, pois um país europeu nunca poderá disputar uma Copa América. Este é um problema apresentado nesta análise. (AGUIAR, 2013, citado por SILVA, 2014).

Com base nos tópicos apresentados, é possível compreender a estrutura com a qual um analisador de linguagem natural pode funcionar, dado um contexto em que a comunicação necessária é providenciada.

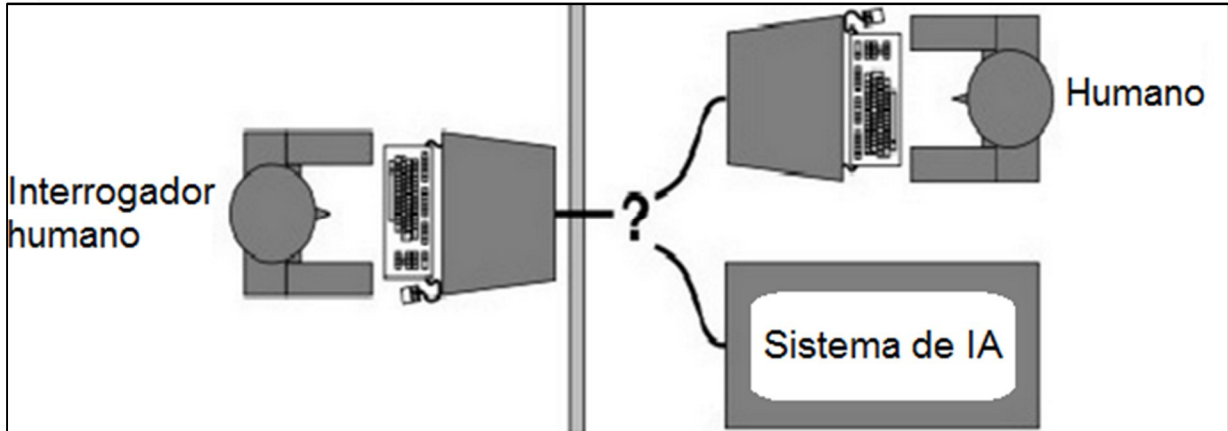
7.4.1 O teste de Turing

A capacidade de se comunicar como um ser humano através do processamento de línguas naturais acaba por trazer um questionamento válido sobre o assunto que é, até os dias de hoje, estudado e debatido pelos profissionais da área: É possível fazer com que um indivíduo artificial se comunique com um ser humano de forma plena?

De acordo com Russel e Norvig (1995, citado por Rosa, 2011), “O teste de Turing [...] foi projetado para fornecer uma definição operacional satisfatória de inteligência.”.

O teste tem como propósito categorizar sistemas como inteligentes a partir da premissa de que tal sistema consiga interagir com um ser humano através de um terminal se passando por outro ser humano. A partir do ponto em que o interrogador não sabe discernir entre quem é o indivíduo artificial e o ser humano baseando-se nas respostas dadas a suas perguntas, é considerado por Turing que o sistema possui inteligência. O teste elaborado por ele se encontra ilustrado na Figura 13.

Figura 13 – O teste de Turing.



Fonte: Rosa (2011, p. 138).

7.4.2 O modelo bag-of-words

O modelo de representação *bag-of-words* possui uma premissa relativamente simples de ser compreendida e implementada. Sua representação parte de um conjunto que possui as características determinísticas para a classificação de um determinado objeto através da avaliação de seus aspectos.

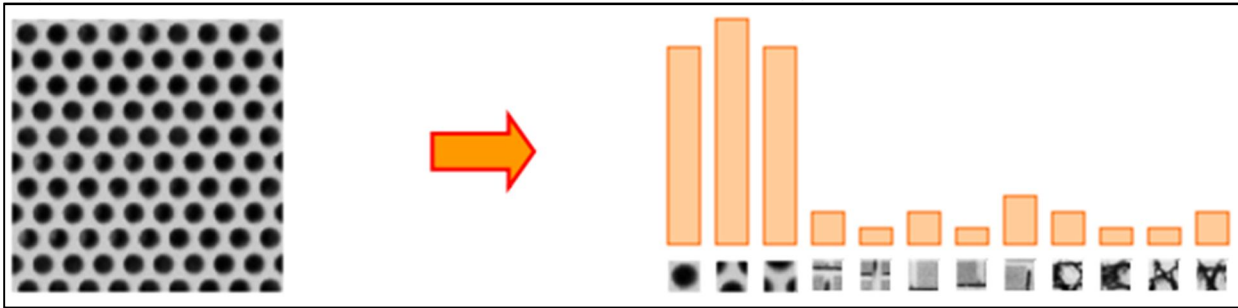
Martins, Matsubara e Monard (2003, p.3) caracterizam o modelo *bag-of-words* aplicado no reconhecimento textual pela representação de um vetor de termos classificados por sua ocorrência em um determinado texto (ou, em sistemas mais sofisticados, por n-agrupamentos de termos).

Os autores ainda definem como etapas essenciais à elaboração de uma *bag-of-words* a coleta de documentos pertinentes à temática abordada, a etapa de pré-processamento na qual os documentos devem se encaixar ao formato da ferramenta utilizada para as etapas posteriores, a extração do conhecimento e, por fim, a avaliação e interpretação dos resultados.

Essas etapas fundamentais descrevem o núcleo do modelo tratado, pois estão presentes em qualquer esquema que se refere a uma *bag-of-words*.

A Figura 14 demonstra a extração e análise que pode ser realizada partindo-se de um objeto de estudo e a aplicação deste modelo. Partindo-se desta perspectiva, é possível compreender como funciona a formulação de uma *bag-of-words*.

Figura 14 – Reconhecimento de textura pelo modelo bag-of-words.



Fonte: Fei-Fei et al. (2012).

Como pode ser observado, é possível extrair os aspectos de maior repetição que caracterizam a Figura 14, e utilizar estes aspectos como ferramenta de avaliação para determinar a similaridade com a imagem inicial.

7.5 ONTOLOGIA

A ontologia pode ser vista como um meio para a organização de informações a partir de uma estrutura estabelecida. Assim, é possível criar o relacionamento entre os componentes dessa estrutura que viabilizam a identificação e a classificação da informação.

Ontologia é uma técnica de organização de informações que vem recebendo especial atenção nos últimos anos, principalmente no que diz respeito à representação formal de conhecimento. Geralmente criadas por especialistas, tendo sua estrutura baseada na descrição de conceitos e dos relacionamentos semânticos entre eles, as ontologias geram uma especificação formal e explícita de uma conceitualização compartilhada. (MORAIS; AMBRÓSIO, 2007, p. 1).

A definição dada sobre o tema contextualiza o significado do termo Ontologia no âmbito computacional, a partir do qual a Inteligência Artificial se beneficia, dando a ela a possibilidade de desbravar uma área de estudo distinta de tudo o que existia até então. No entanto, o estudo da ontologia não surgiu no ambiente da computação, mas sim muito antes como uma das discussões filosóficas mais conhecidas na antiguidade.

Segundo Ambrósio e Moraes (2007), a ontologia em seu sentido filosófico possui diversas definições. Analisando a palavra, trata-se de uma derivação do grego *onto*, ser, e *logos*, palavra.

Embora não exista uma definição final para o termo subjetivo ontologia no meio filosófico, é possível compreender seu fundamento essencial com facilidade, como é definido por Alcadipani e Cavalcanti (2010, p. 2), a ontologia sendo “[...] o fundamento por meio do qual se torna possível construir uma teoria para se conhecer uma dada realidade.”. Os autores ainda referenciam Fleetwood (2005) que reforça esse ponto de vista quando ele afirma que se trata de um componente essencial da dinâmica do saber organizacional.

Para a Ciência da Informação, o desenvolvimento de ontologias surge da necessidade da representação da informação em meio digital, contexto que permeia a produção, a disseminação e a recuperação dos conteúdos na atualidade. (GONÇALVES; SOUZA, 2008, p. 1).

No contexto científico, fica clara a importância da Ontologia pelo potencial de gerenciamento de informação que há. Alguns autores como Dahlberg (2006) e Hjørland (2007) relacionam esta área introduzindo um novo termo, a Organização do Conhecimento.

Sobre Organização do Conhecimento, Dahlberg (2006) citado por Gonçalves e Souza (2008?, p. 2) define que se trata de uma ciência que “estrutura e arranja sistematicamente unidades do conhecimento (os conceitos) de acordo com os elementos de conhecimento (características) comuns.”.

Bastos, Moreno, Nascimento e Oliveira (2007) citam Gruber (1993) ao afirmar que a Ontologia se trata na verdade de “definições explícitas de conceitos e suas relações”, o que remete ao fato de que ela pode ser aplicada em diversas áreas do conhecimento conforme se ajusta ao contexto em questão, tal como pode ser observado na Figura 15.

Figura 15 – Aplicações da ontologia.

Aplicações da ontologia	
ÁREA DE APLICAÇÃO	DEFINIÇÃO
Inteligência Artificial	GUARINO (1997) define ontologia como uma caracterização axiomática do significado do vocabulário lógico
Sistemas de Informação	E definida como um conjunto de conceitos e termos que podem ser usados para descrever alguma área do conhecimento ou construir uma representação para o conhecimento (SWATOUT, 1999)
Linguagem e Cognição	A ontologia refere-se a tudo que existe no mundo composto por objetos, mudanças e relações entre eles. Pode ser baseada no mundo, na mente/intelecto, na cultura ou na linguagem (DAHLGREN, 1995).
Banco de Dados	Conhecimento genérico que pode ser reusado em aplicações de tipos diferentes (MEERSMAN, 2002)

Fonte: Vitorino e Bräscher (2006, citado por Bastos, Moreno, Nascimento e Oliveira, 2007, p. 14).

Bezerra et al. (2006, p. 568) ainda contribuem com o assunto ressaltando o fato de que deve existir um vocabulário específico que pertence a algum dos conjuntos da ontologia para o domínio estabelecido.

Campos et al. (2005, citado por CAMPOS e MARCONDES, 2008, p. 112) complementam essa linha de raciocínio com o fato de que não somente um conjunto vocabulário existe para um determinado domínio dentro de uma ontologia, como também a relação lógica entre as unidades que fazem parte deste conjunto. Desta forma, é possível estabelecer a ligação entre essas unidades em uma ontologia formal.

7.6 TAXONOMIA

A taxonomia pode ser entendida como a ciência que se propõe a classificar indivíduos em seus respectivos grupos partindo de suas características pertinentes à classificação.

Novo (2010, p. 135) introduz taxonomia em seu artigo contextualizando-a historicamente. De acordo com a autora, o termo se popularizou no século XVIII no domínio da Biologia quando as pesquisas do biólogo Carolus Linnaeus culminaram na classificação hierárquica de um Reino em filos, ordens, famílias, gênero e espécie que, posteriormente, ainda foi subdividida e conhecida como Taxonomia de Lineu.

As taxonomias atualmente são estruturas classificatórias que têm por finalidade servir de instrumento para a organização e recuperação de informação nas empresas. Estão sendo vistas como meios de acesso atuando como mapas conceituais dos tópicos explorados em um serviço de recuperação. O desenvolvimento de taxonomias para o negócio da empresa tem sido um dos pilares da gestão da informação e do conhecimento. (volume de informação requer padronização). (BAILEY, K. 2007; GILCHRIST, A. 2003; OPDAHL, A. L. & SINDRE, G. 1994, citado por CAMPOS e GOMES, 2007).

A definição mencionada por Campos e Gomes (2007) confirma a importância desta ciência para um mundo em que a informação é analisada e extrapolada com diversos propósitos.

Os autores ainda ressaltam que taxonomias não podem ser consideradas neutras no sentido de que, embora sejam flexíveis ao ponto de que possam ser aplicadas em qualquer tipo de contexto, também devem ser consideradas parciais ao contexto. Assim, uma determinada taxonomia sempre carregará as características da organização sobre a qual ela foi desenvolvida.

Novo (2010, p. 131) caracteriza taxonomia como uma estrutura classificatória que existe com a finalidade de permitir agregação de dados e, posteriormente, de informação.

8 TRABALHOS CORRELATOS

Após ampla pesquisa que se deu início assim que a ideia de desenvolver este sistema surgiu, foram encontrados diversos projetos que, embora atuem em áreas diferentes e/ou fazendo uso de métodos diferentes, podem ser relacionados com o núcleo principal que é tratado neste sistema. A seguir, são apresentadas as produções relevantes que podem ser relacionadas a este trabalho.

Carvalho, C. L. e Santos (2008) produziram um artigo com o propósito de tratar sobre as ferramentas de Inteligência Artificial que poderiam auxiliar no gerenciamento de conhecimento. Em seus levantamentos foram estabelecidos como recursos de apoio o uso de redes neurais, sistemas especialistas e agentes inteligentes. Leva-se em conta que ao falar sobre a administração de conhecimento, os autores se referem neste artigo sobre, de forma geral, mecanismos inteligentes de busca.

O artigo de Cardoso e Machado (2008) tem como objetivo tratar da gestão do conhecimento relacionado às técnicas de mineração de dados – disciplina multidisciplinar que envolve desafios de banco de dados e Inteligência Artificial muito estudada nos dias de hoje. É interessante sua visão sobre o assunto, pois em seu artigo a gestão de conhecimento é tratada de forma muito parecida como ela é abordada neste projeto. Trata-se de um recurso de extrema importância para as empresas nos dias de hoje.

Zaidan (2010?) trata das formas como o conhecimento pode ser refinado partindo do ponto de vista da engenharia de *software*. Segundo o autor, é possível extrair uma quantidade maior de valor sobre um conhecimento desde que este seja administrado de maneira apropriada.

Botelho (2015) dissertou sobre o assunto “Aplicações de ontologias na organização de conteúdos para apoio a equipes de desenvolvimento de *software*”. Em sua produção, o autor trata da necessidade de melhorar a velocidade e relevância da informação que é buscada pelo desenvolvedor de *software*.

Almeida et al. (2005) produziram um artigo que trata puramente sobre o desenvolvimento de um sistema de gestão de conhecimento e seus objetivos foram de

contribuir com o cenário brasileiro da gestão de conhecimento como uma área que, desde então, tem passado por ascensão e gerado cada vez mais interesse do mercado.

Werner e Werner (2004) são autores que falam sobre meios de manipular e aplicar a gestão de conhecimento em empresas de pequeno e médio porte brasileiras. A visão administrativa desses autores acaba por demonstrar que os meios para o gerenciamento do conhecimento podem ser muito mais amplos que um sistema dedicado, como ferramentas de apoio sendo um dos exemplos tratados em sua produção.

É interessante observar os meios, técnicas e objetivos sobre os quais foram fundamentadas as pesquisas de todos os autores acima. Tendo em perspectiva todas essas variações, entende-se que a disciplina de gestão de conhecimento é conveniente de ser aplicada em qualquer área que seja passível de gerenciamento.

Em adição, com base no que foi pesquisado, conclui-se que, embora existam diversas produções que abordem tanto a gestão de conhecimento quanto a ontologia com o propósito de organização empresarial, existe espaço para pesquisa na área e diversas formas de se abordar o assunto tratado, além de contextos em que pode ser aplicado. Com isso, entende-se que o mercado brasileiro de gestão de conhecimento ainda é jovem e deve ser bastante explorado.

9 METODOLOGIA

Este trabalho teve como finalidade a criação de um sistema de gestão de conhecimento com base em uma ontologia com o propósito de proporcionar uma alternativa para as etapas organizacionais de uma empresa.

Para tal, a princípio, foram realizadas extensas pesquisas exploratórias que visaram melhor compreensão sobre o que é a gestão de conhecimento, como ela poderia ser aplicada em um ambiente corporativo e como as ferramentas administrativas de uma empresa funcionam, tendo em perspectiva o ponto de vista dos usuários, dos investidores e dos desenvolvedores de tais aplicações.

De acordo com Mattar (2012), uma pesquisa exploratória ajuda o pesquisador, a saber, quais das várias opções se aplicam ao problema de pesquisa. Além disso, poderá também auxiliar a estabelecer as prioridades a pesquisar. As prioridades poderão ser estabelecidas porque uma particular hipótese explicativa surgida durante a pesquisa exploratória parecerá mais promissora do que outras. Além do mais, a pesquisa exploratória poderá gerar informações sobre as possibilidades práticas da condução de pesquisas específicas.

Essas pesquisas foram realizadas através da leitura de artigos, periódicos, livros, revistas científicas e dissertações produzidas por profissionais, acadêmicos e pesquisadores atuando sobre a área de gestão de conhecimento.

9.1 FERRAMENTAS UTILIZADAS

A seguir, serão apresentadas as ferramentas que foram utilizadas durante o desenvolvimento do projeto considerando todas as suas etapas.

9.1.1 Hardware

Seguindo a estrutura estabelecida, a aplicação foi desenvolvida utilizando como ferramenta principal um notebook da marca Acer quad-core com processadores da família i5 de 2.3GHz e 4Gb de memória RAM utilizando o sistema operacional Windows

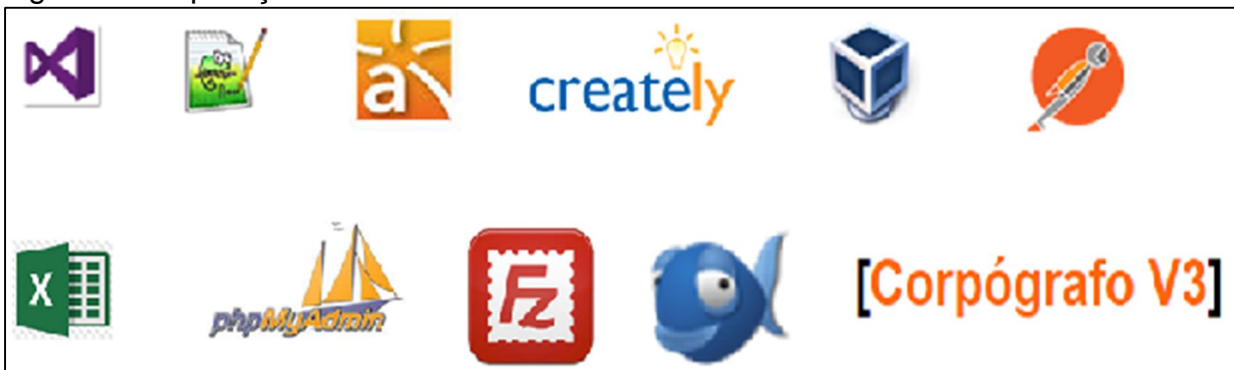
10 - 64 bits. Como ferramenta de apoio, foi utilizado um notebook da marca Sim+ dual-core com processadores de 1.8GHz e 2Gb de memória RAM utilizando o sistema operacional Windows 7 Professional – 64 bits. Este aparelho auxiliar foi utilizado no decorrer do projeto com o propósito de testar compatibilidades em função do sistema desenvolvido.

A escolha de ambos os aparelhos se dá pelo fato de que são propriedades do autor, em adição a terem sido o suficiente para suprir todas as necessidades que surgiram no decorrer do desenvolvimento, além do que já foi estabelecido.

9.1.2 Softwares

Quanto ao nível de *software*, foram utilizadas ferramentas de apoio ao desenvolvedor, complementos que facilitam o teste de *software*, programas de modelagem UML e gerenciadores de banco de dados, tratados a seguir de acordo com a Figura 16.

Figura 16 – aplicações utilizadas durante o desenvolvimento.



Fonte: elaborado pelo autor.

- a) **Microsoft Visual Studio 2013**: plataforma escolhida em função de ter sido justamente lançada com o propósito de proporcionar uma experiência simples, objetiva e eficiente aos desenvolvedores que optam por ela. Em adição a ser uma ferramenta gratuita, ela também conta com todas as ferramentas que poderiam vir a ser necessárias no decorrer do projeto;

- b) **Notepad++**: foi utilizado com o objetivo de servir como apoio em todo o código de qualquer linguagem de programação. Esta ferramenta gratuita é interessante por reconhecer uma gama de linguagens e permitir a compreensão de sintaxes e semânticas a partir da mesma;
- c) **Astah Community**: esta ferramenta oferece ao programador uma forma simples e rápida de desenvolver diagramas de modelagem UML. A escolha pela ferramenta se dá ao fato de ser disponibilizada gratuitamente pela própria empresa Astah, além de ter sido utilizada em sala de aula durante o aprendizado sobre a modelagem de diagramas no decorrer da disciplina de Engenharia de *Software II*;
- d) **Creately**: esta ferramenta auxiliar a modelagem do sistema é uma aplicação web que permite o desenvolvimento de diversos diagramas. Além de ser gratuita, é bastante versátil e foi utilizada em complemento à Astah Community;
- e) **Oracle VM VirtualBox**: este *software* permite a emulação de um sistema operacional a partir da criação de uma máquina virtual. Foi utilizada no notebook auxiliar que serviu como servidor LAMP (Linux – Ubuntu Desktop 14.0 –, Apache, MySQL, PHP);
- f) **Postman**: este complemento para o navegador Chrome permite que sejam realizados envios de parâmetros (*request*) e recebimento de respostas (*response*) para um determinado endereço. A ferramenta se mostra útil partindo do princípio de que todos os scripts PHP devem ser testados antes de serem utilizados pelo sistema;
- g) **Microsoft Excel 2013**: a ferramenta foi utilizada com o objetivo de criar esboços das tabelas do banco de dados do sistema e elaborar diagramas/levantamentos estatísticos sobre aspectos diversos do projeto. Sua escolha é justificada pelo fato de oferecer os recursos necessários de forma simples e objetiva para preencher esta necessidade do projeto;
- h) **Phpmyadmin**: o PHPMyAdmin é um serviço web criado em PHP que possibilita a criação e manipulação de bases de dados. Hoje, este SGBD é um dos mais populares quando se trata de bancos de dados MySQL, junto ao sistema MySQL Workbench. A escolha é justificada pelo fato de que o serviço é simples,

funcional e por oferecer todas as ferramentas que serão necessárias em seu contexto;

- i) **Filezilla Server:** este *software* foi utilizado com o propósito de configurar o aspecto File Transfer Protocol (FTP) do servidor. A escolha por este *software* é dada pelo fato de que se trata de um modelo simples e ágil de se gerenciar o aspecto FTP;
- j) **Bluefish:** este *software* é um editor de documentos PHP que foi utilizado no decorrer da elaboração dos *scripts* de interação com o banco de dados no servidor. Sua escolha é justificada por ser uma ferramenta gratuita e popular no meio de desenvolvimento Linux;
- k) **Corpógrafo:** esta ferramenta web tem como propósito realizar análises linguísticas sobre um determinado texto ou conjuntos textuais para extrair diversas informações que podem ser utilizadas para fins estatísticos ou levantamentos de informação. Ao contexto deste projeto, foi utilizada para extrair de grandes quantidades de palavras os termos pertinentes mais recorrentes.

9.2 ESTRUTURA DO SERVIDOR

Em relação à estrutura do servidor, foi configurada uma máquina virtual do tipo LAMP com o propósito de conter o banco de dados MySQL, os *scripts* PHP e o serviço de aplicação FTP que serviram como interação entre a aplicação e a base de conhecimento, além de armazenar os arquivos do usuário na aplicação.

A máquina virtual teve sua placa de rede configurada no formato BRIDGE com o propósito de estabelecer fácil comunicação com o sistema em qualquer dispositivo conectado à mesma rede sem fio.

A escolha do sistema operacional Linux Ubuntu Desktop versão 14.0 para a conclusão do projeto em nível de servidor se dá ao fato de que este sistema já foi utilizado pelo autor anteriormente com a finalidade de criar um sistema LAMP sem que tivessem sido apresentadas quaisquer dificuldades, desde a implementação até o próprio desenvolvimento.

Os *scripts* PHP foram criados fazendo uso da ferramenta Bluefish, enquanto o banco de dados foi esboçado utilizando a ferramenta de *software* Microsoft Excel 2013, expressando a estrutura de todas as tabelas, assim como a relação que existe entre elas e uma breve documentação sobre cada uma dessas tabelas.

9.3 LINGUAGENS DE PROGRAMAÇÃO E MODELAGEM

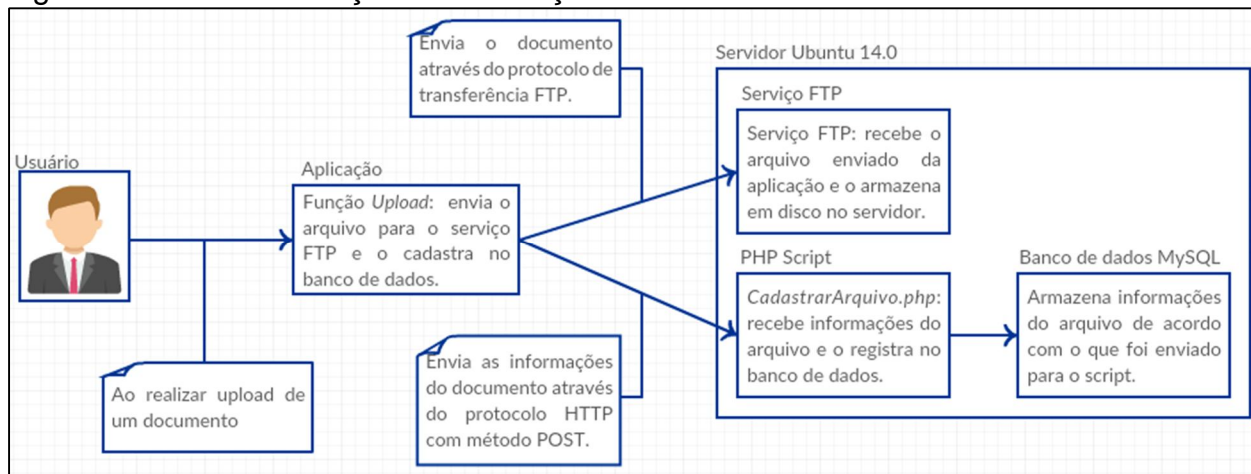
As linguagens de programação utilizadas neste projeto foram implementadas de forma a complementarem uma a outra pelo bem da aplicação.

- a) **A UML:** o padrão UML foi a primeira linguagem a ser utilizada e desempenhou o papel de modelar a estrutura do sistema, sobre a qual o mesmo foi desenvolvido. Assim, trata-se de uma escolha fundamental para as etapas seguintes que se apoiam em como o sistema funcionaria. A linguagem de modelagem foi aplicada com o *software* Astah Community para criar os Diagramas de Casos de Uso, de Classes e de Atividades;
- b) **A linguagem SQL:** foi utilizada na criação do banco de dados e suas tabelas, além do gerenciamento desses dados quando forem solicitados por funções que requerem manipulação do banco de dados;
- c) **A linguagem PHP:** foi utilizada com o propósito de manipular o banco de dados MySQL da aplicação a partir de *scripts* que foram desenvolvidos com propósitos individuais e específicos. Sua escolha é justificada pelo fato de que o PHP faz parte do pacote LAMP e já ser consolidado no meio de desenvolvimento;
- d) **A linguagem C#:** foi utilizada com o propósito de criar toda a estrutura e lógica de programação do sistema que foi desenvolvido. O C# foi escolhido por ser uma linguagem moderna e crescente no mercado. Além de oferecer todos os recursos que serão necessários a nível de programação, C# também possui excelente suporte e uma vasta comunidade de desenvolvedores disponíveis para consulta.

Como pode ser observado, existe uma relação entre as linguagens escolhidas, a qual é fundamental para o bom funcionamento do sistema. Essa relação pode ser mais bem compreendida a partir do diagrama da Figura 17, no qual um usuário interage com

o sistema de forma a realizar *upload* de um documento no servidor, acionando uma cadeia de eventos e interações entre o sistema e o servidor.

Figura 17 – Cadeia de ações na interação com o sistema.



Fonte: elaborada pelo autor.

9.4 MODELAGEM DO SISTEMA

Em nível de modelagem, foram utilizadas as ferramentas Creately – para modelagem de apoio e de diagramas utilizados no projeto – e Astah Community, com o propósito de expressar a estrutura do sistema, tal como de elaborar o planejamento da parte prática do trabalho.

Esses diagramas foram posteriormente utilizados com fins de consulta no decorrer da implementação da ideia inicial.

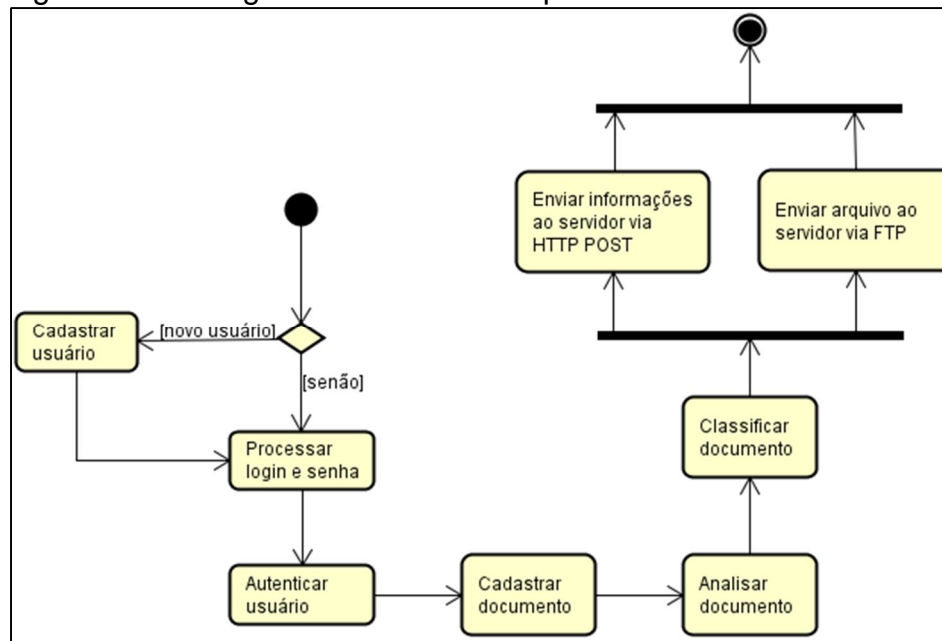
Com o propósito de cobrir todas as áreas pertinentes deste projeto foram utilizados os diagramas de Atividades, Casos de Uso e de Classes.

9.4.1 Diagramas de Atividades

A seguir, nas Figuras 18, 19, 20 e 21, são tratados os Diagramas de Atividades expressando os procedimentos pertinentes à temática do projeto desenvolvido.

A Figura 18 demonstra o conjunto de procedimentos envolvidos no processo de inclusão de um determinado documento no sistema (e, por consequência, no servidor).

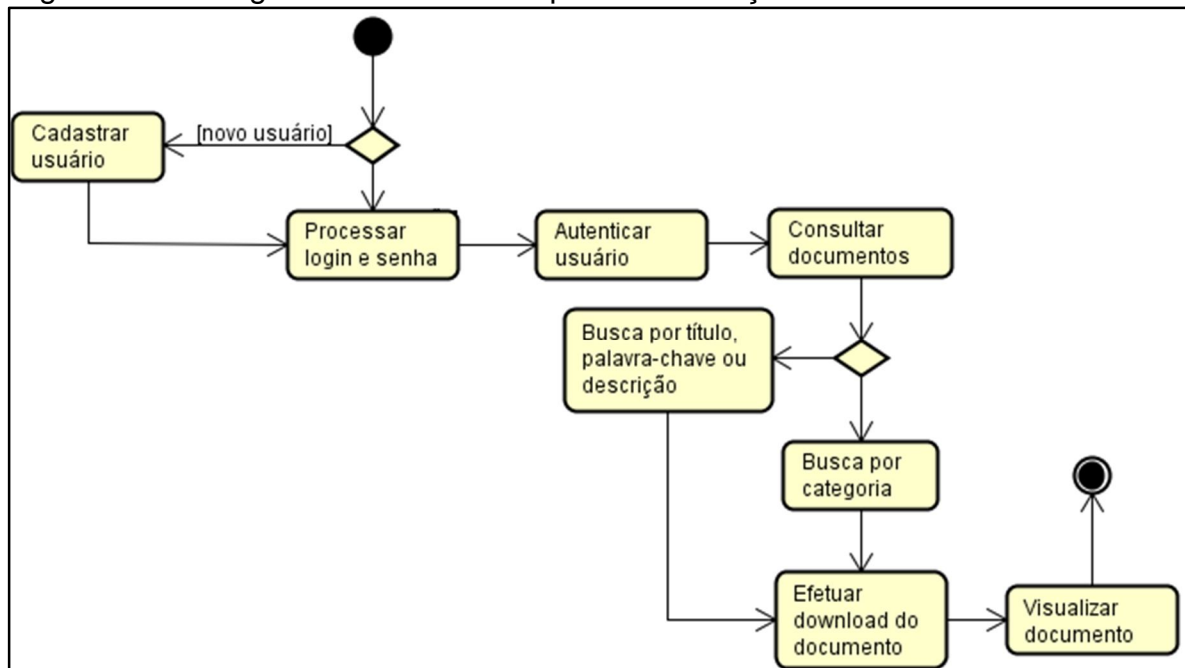
Figura 168 – Diagrama de Atividades para inclusão de documentos.



Fonte: elaborada pelo autor.

A Figura 19 demonstra o processo de visualização de um determinado arquivo que foi inserido no servidor, desde a ação de autenticar-se como usuário daquele sistema, até o *download* e a visualização do arquivo que foi selecionado.

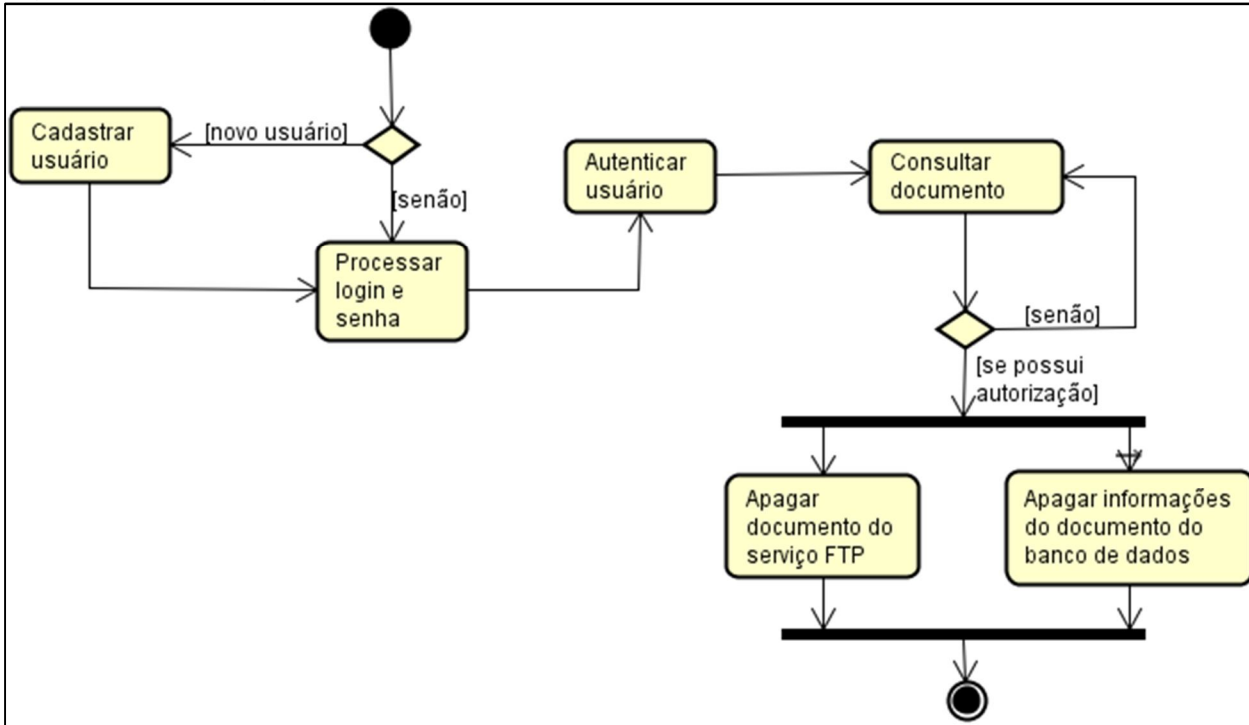
Figura 179 – Diagrama de Atividades para visualização de documentos.



Fonte: elaborada pelo autor.

A Figura 20 demonstra o procedimento de exclusão de um arquivo que foi previamente cadastrado no sistema. Conforme será abordado no tópico 9.5.2 – busca de documentos, é importante ressaltar que um usuário comum do sistema não possui nativamente a permissão necessária para realizar este procedimento.

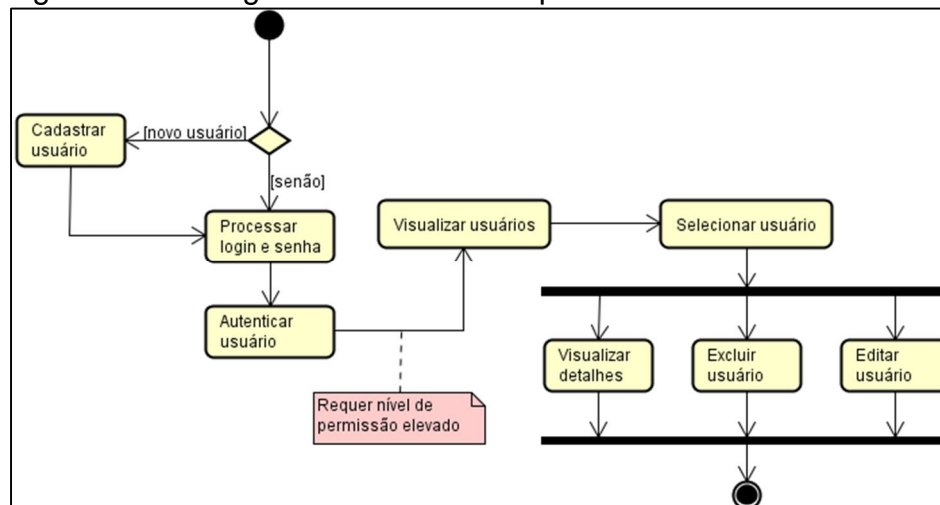
Figura 2018 – Diagrama de Atividades para exclusão de documentos.



Fonte: elaborada pelo autor.

A Figura 21 demonstra os procedimentos disponíveis para um usuário com nível de permissão superior em função dos outros usuários cadastrados no sistema. Assim, o sistema foi programado de tal forma que qualquer membro com acesso a este módulo pode aumentar a permissão de outros membros da empresa.

Figura 191 – Diagrama de atividades para manter usuários.

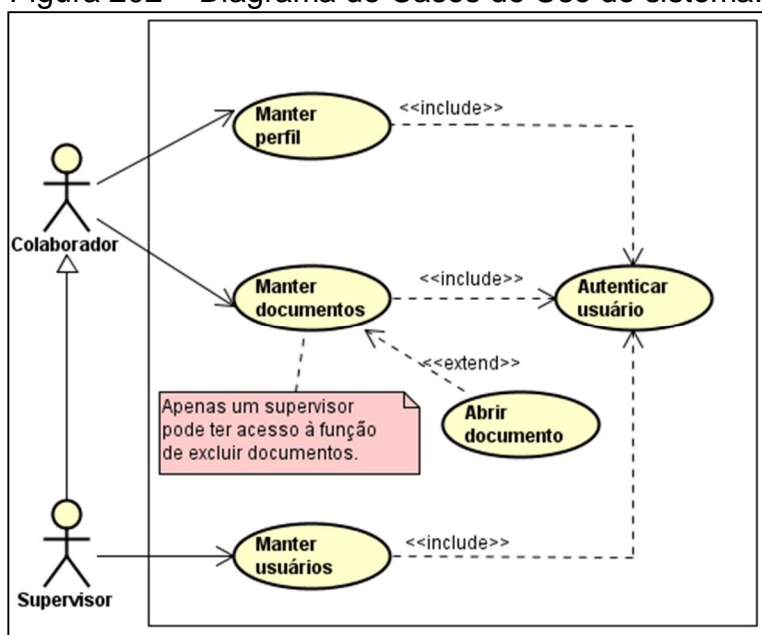


Fonte: elaborada pelo autor.

9.4.2 Diagrama de Casos de Uso

A seguir, na Figura 22 será abordado o Diagrama de Casos de Uso que foi desenvolvido com o propósito de demonstrar as funções principais do sistema que foi desenvolvido. Este diagrama trata das funções que podem ser desempenhadas por um indivíduo cadastrado no sistema, de acordo com seu respectivo nível de acesso.

Figura 202 – Diagrama de Casos de Uso do sistema.

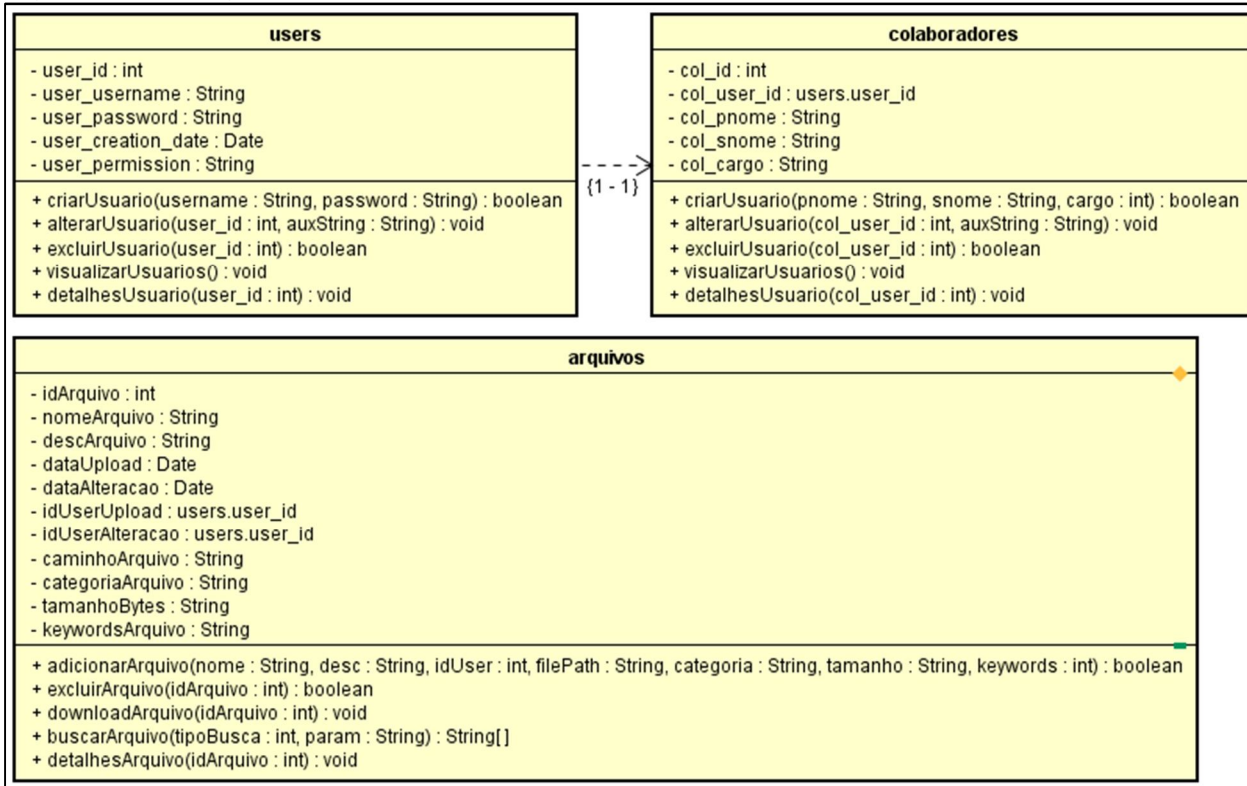


Fonte: elaborada pelo autor.

9.4.3 Diagrama de classes

A seguir, na Figura 23, é apresentado o conjunto de classes e funções a nível de programação do sistema.

Figura 23 – Diagrama de Classes do sistema.



Fonte: elaborada pelo autor.

A Figura 23 demonstra as classes, assim como seus atributos e métodos. Como pode ser visualizado, existe uma relação clara entre as classes “users” e “colaboradores”, sendo que o atributo “col_user_id” herda como chave estrangeira o atributo “user_id”, estabelecendo, portanto uma ligação entre as duas classes.

9.5 ELABORAÇÃO DA ONTOLOGIA

Em paralelo ao desenvolvimento do projeto, a linguagem C# também foi utilizada para expressar a ontologia que foi criada com o propósito de classificar o conhecimento tratado pelo sistema. Essa ontologia se encontra intrínseca ao próprio sistema, o que viabilizou a utilização do C# como escolha apropriada neste contexto.

A seguir, são abordadas as etapas envolvidas em todo o processo de elaboração da ontologia.

9.5.1 Estudo exploratório de modelos disponíveis para implementação

O procedimento escolhido foi a metodologia *bag-of-words* que serve apropriadamente à implementação na linguagem de programação escolhida. A seleção do modelo *bag-of-words*, por sua vez, é dada por ser um conceito relativamente simples com histórico de bons resultados e por ter demonstrado agilidade a tempo de execução e processamento dos documentos utilizados na fase de testes.

9.5.2 Extração de palavras pertinentes ao contexto

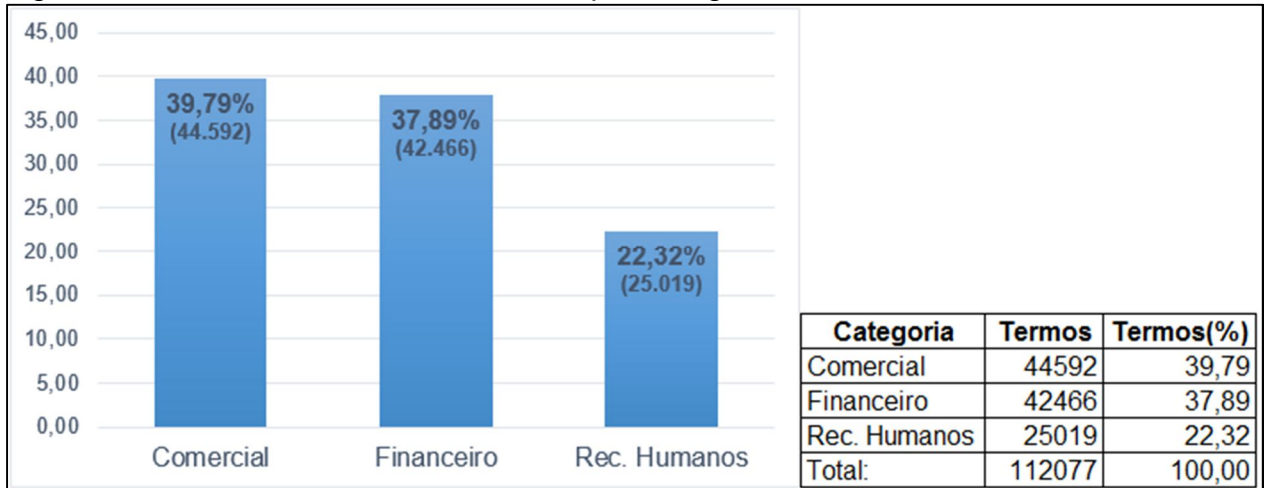
A etapa seguinte da elaboração da ontologia começou com a coleta de um grande volume de textos, artigos, documentos e periódicos sobre as temáticas propostas para implementação (Comercial, Financeiro e Recursos Humanos) equivalentes a um total de 112077 termos.

Esses textos foram então avaliados com o propósito de validar seu conteúdo como pertinente à sua respectiva temática. Ao final da fase de avaliação dos documentos coletados, o conteúdo foi indiscriminadamente unido em um mesmo documento de texto para cada uma das categorias com o propósito de ser processado pela ferramenta Corpógrafo, gerando, portanto, três conjuntos de textos, artigos, documentos e periódicos representando as categorias Comercial, Financeiro e Recursos humanos preparados para a análise da ferramenta.

Os três arquivos finais foram copiados para o servidor da ferramenta Corpógrafo a partir da qual foram realizadas as extrações de n-termos (termos compostos por n palavras, dada a variação $4 > n > 0$).

Os resultados brutos dessa extração podem ser vistos na Figura 24.

Figura 24 – Total de termos encontrados por categoria.



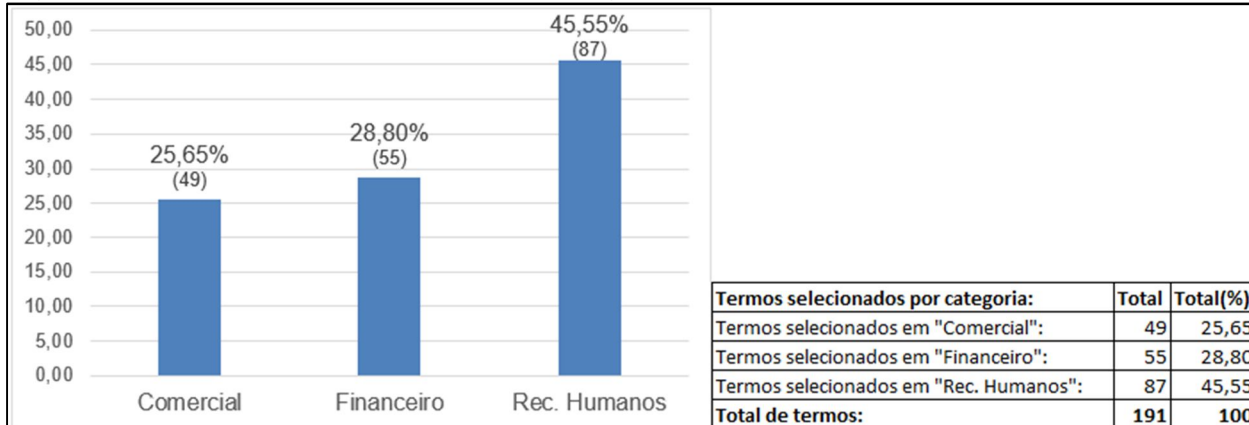
Fonte: elaborada pelo autor.

Conforme pode ser visualizado na Figura 24, foi levantado um total de 112077 termos nesta etapa, a partir dos quais foi realizada uma avaliação e seleção dos termos pertinentes a cada um dos três contextos propostos.

9.5.3 Avaliação e seleção dos termos levantados

A próxima etapa na elaboração da ontologia foi realizada com base nos termos apresentados pelo Corpógrafo utilizando as opções de extração de n-termos, tal que $4 > n > 0$. Nesta fase, houve uma seleção arbitrária dos termos que poderiam ser considerados pertinentes a uma determinada categoria do sistema. A Figura 25 apresenta os resultados desta seleção.

Figura 25 – Termos selecionados por categoria.

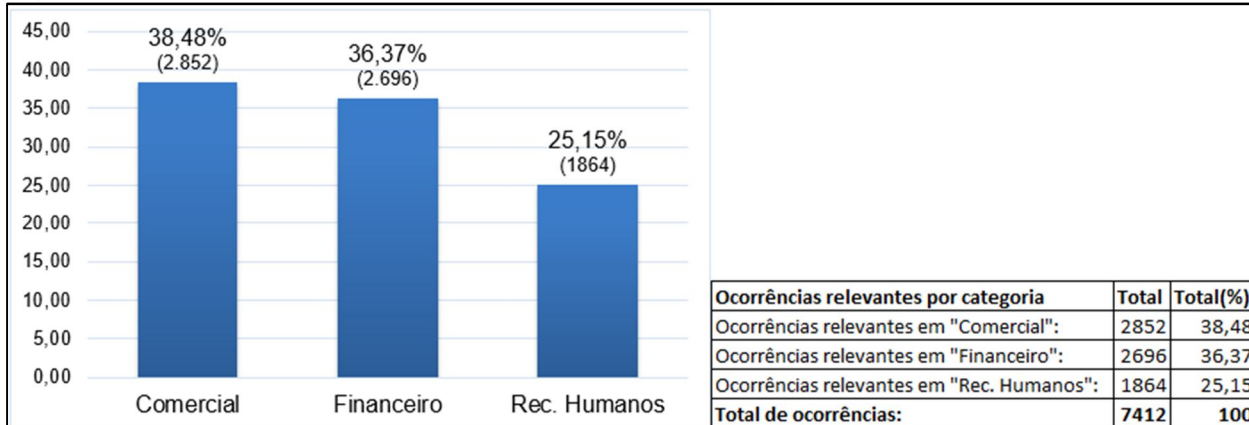


Fonte: elaborada pelo autor.

A Figura 25 demonstra o número de termos que foram selecionados pelo autor para a elaboração da ontologia.

É interessante de se ressaltar que, embora a categoria “Recursos Humanos” esteja atribuída ao maior número de termos distintos pertinentes a sua classe, a Figura 25 demonstra que esta categoria também recebeu o menor número de ocorrências desses 87 termos, enquanto “Comercial” possui uma quantidade inferior de termos selecionados, porém o maior número de ocorrências em seus 49 termos. Essa análise sobre o vocabulário levantado permite a interpretação de que a área empresarial de Recursos Humanos da empresa analisada trata de assuntos mais abrangentes, enquanto o setor Comercial pode ser considerado um nicho mais específico dentro da organização.

Figura 26 – Ocorrência de termos relevantes por categoria.



Fonte: elaborada pelo autor.

Como foi abordado anteriormente, embora o vocabulário da categoria Comercial tenha sido inferior ao vocabulário das demais categorias, este setor possui a maior repetição de termos, enquanto a classe com quantidade de termos mais abrangente possui o menor número de repetições. O setor Financeiro, no entanto, se encontra entre os demais em ambos gráficos das Figuras 26 e 25.

A seguir, são listados os n-termos extraídos do levantamento feito pela ferramenta Corpógrafo e o número de ocorrências de cada um desses n-termos para as categorias Comercial, Financeiro e Recursos Humanos, respectivamente.

A Figura 27 ilustra os termos selecionados para a categoria **COMERCIAL**.

Figura 27 – Termos selecionados para a categoria Comercial.

#	Palavras	Ocorrência	Ocorrência(%)	#	Palavras	Ocorrência	Ocorrência(%)
1	clientes	263	9,22	26	concorrência	14	0,49
2	comercial	309	10,83	27	necessidade	14	0,49
3	produtos	225	7,89	28	procedimento	11	0,39
4	empresa	616	21,60	29	atender, atendimento	10	0,35
5	serviços	58	2,03	30	gerente	6	0,21
6	marketing	86	3,02	31	negócio	28	0,98
7	tecnologia	8	0,28	32	consumidor, consumo	50	1,75
8	equipe	64	2,24	33	preço	30	1,05
9	unidade	3	0,11	34	comprador	10	0,35
10	mercado	145	5,08	35	fornecedor	13	0,46
11	informações	17	0,60	36	gestão comercial	28	0,98
12	conta	2	0,07	37	representante comercial	31	1,09
13	processos	7	0,25	38	departamento comercial	24	0,84
14	controle	4	0,14	39	planejamento estratégico	24	0,84
15	ganho	1	0,04	40	gestor comercial	22	0,77
16	planejamento	111	3,89	41	representação comercial	21	0,74
17	terceirizar	5	0,18	42	setor comercial	17	0,60
18	vendas	380	13,32	43	equipe comercial	15	0,53
19	vendedor	102	3,58	44	gerente comercial	14	0,49
20	público	1	0,04	45	gestão econômica	7	0,25
21	sistema	2	0,07	46	atividade comercial	5	0,18
22	solução	2	0,07	47	produtos e/ou serviços	12	0,42
23	vantagem	2	0,07	48	equipe de vendas	7	0,25
24	desempenho	1	0,04	49	atendimento ao cliente	5	0,18
25	atendimento	20	0,70				

Fonte: elaborada pelo autor.

A Figura 27 ilustra os termos selecionados para a categoria **FINANCEIRO**.

Figura 28 – Termos selecionados para a categoria “Financeiro”.

#	Palavras	Ocorrência	Ocorrência(%)	#	Palavras	Ocorrência	Ocorrência(%)
1	capital	353	13,09	29	crédito	38	1,41
2	empresa	345	12,80	30	pagamento, pagar	91	3,38
3	valor	135	5,01	31	orçamento	11	0,41
4	lucro	83	3,08	32	prazo	24	0,89
5	retorno	69	2,56	33	balanço	17	0,63
6	resultado	46	1,71	34	gasto	17	0,63
7	caixa	66	2,45	35	fiscal	18	0,67
8	contábil, contabilidade	40	1,48	36	planejamento financeiro	23	0,85
9	mercado	73	2,71	37	educação financeira	15	0,56
10	econômico	35	1,30	38	administração financeira	15	0,56
11	investimento, investido	73	2,71	39	patrimônio líquido	13	0,48
12	fluxo	50	1,85	40	lucro econômico	11	0,41
13	taxa	30	1,11	41	lucro residual	11	0,41
14	recurso	71	2,63	42	finanças corporativas	10	0,37
15	líquido	71	2,63	43	gestão financeira	10	0,37
16	riqueza	28	1,04	44	valor econômico	10	0,37
17	desempenho	27	1,00	45	equilíbrio financeiro	10	0,37
18	finanças	187	6,94	46	finanças pessoais	7	0,26
19	custo	21	0,78	47	análise financeira	5	0,19
20	investidor	20	0,74	48	controle financeiro	5	0,19
21	financeira	100	3,71	49	custo do capital	34	1,26
22	remuneração	15	0,56	50	contas a receber	16	0,59
23	avaliação, análise	20	0,74	51	contas a pagar	14	0,52
24	dinheiro	21	0,78	52	nota fiscal	19	0,70
25	empréstimo	10	0,37	53	cupom fiscal	6	0,22
26	fundo	109	4,04	54	folha de pagamento	4	0,15
27	ativos	46	1,71	55	prestação de contas	4	0,15
28	contas	94	3,49				

Fonte: elaborada pelo autor.

As Figuras 29.1 e 29.2 ilustram os termos selecionados para a categoria **RECURSOS HUMANOS**.

Figura 29.1 – Termos selecionados para a categoria “Rec. Humanos”.

#	Palavra	Ocorrência	Ocorrência(%)	#	Palavra	Ocorrência	Ocorrência(%)
1	salário	184	9,87	29	renda	9	0,48
2	empregado	179	9,60	30	emprego	8	0,43
3	férias	109	5,85	31	doença	8	0,43
4	trabalho	168	9,01	32	abono	7	0,38
5	pagamento	56	3,00	33	notificação	7	0,38
6	contrato	38	2,04	34	término	8	0,43
7	rescisão	36	1,93	35	licença	8	0,43
8	empregador	35	1,88	36	certidão	26	1,39
9	afastamento	34	1,82	37	pessoa	5	0,27
10	horas	33	1,77	38	aposentadoria	5	0,27
11	serviço	27	1,45	39	maternidade	5	0,27
12	social	25	1,34	40	gravidez	5	0,27
13	previdência	24	1,29	41	assinatura	57	3,06
14	carteira	24	1,29	42	servidor	36	1,93
15	gozo	20	1,07	43	requerimento	41	2,20
16	aprendiz	31	1,66	44	recebedor	21	1,13
17	inss	20	1,07	45	matrícula	18	0,97
18	admitir	31	1,66	46	funcionário	17	0,91
19	trabalhador	19	1,02	47	cargo	17	0,91
20	folha	19	1,02	48	responsabilidade	16	0,86
21	demissão	22	1,18	49	competência	12	0,64
22	falta, faltas	17	0,91	50	transporte	9	0,48
23	aviso	16	0,86	51	chefe	7	0,38
24	extra	16	0,86	52	treinamento	9	0,48
25	família	16	0,86	53	supervisor	7	0,38
26	profissional	11	0,59	54	gratificação	4	0,21
27	acidente	11	0,59	55	nascimento	4	0,21
28	sindicato, sindical	9	0,48	56	previdência social	22	1,18

Fonte: elaborada pelo autor.

Figura 29.2 – Termos selecionados para a categoria “Rec. Humanos”.

#	Palavra	Ocorrência	Ocorrência(%)	#	Palavra	Ocorrência	Ocorrência(%)
57	avaliação global	15	0,80	73	rescisão antecipada	3	0,16
58	rescisão contratual	14	0,75	74	gozo das férias, de férias	16	0,86
59	menor aprendiz	12	0,64	75	domingo	19	1,02
60	salário família, família	12	0,64	76	domingo ou feriado	8	0,43
61	justa causa	12	0,64	77	adicional de insalubridade	9	0,48
62	verbas rescisórias	11	0,59	78	adicional de periculosidade	6	0,32
63	contribuição sindical	9	0,48	79	concessão das férias	6	0,32
64	serviço militar	9	0,48	80	jornada de trabalho	6	0,32
65	férias proporcionais	7	0,38	81	pagamento dos empregados	6	0,32
66	trabalho semanal	6	0,32	82	certidão de casamento	6	0,32
67	salário proporcional	5	0,27	83	folha de ponto	5	0,27
68	periculosidade	7	0,38	84	certidão de nascimento	4	0,21
69	auxílio transporte	3	0,16	85	contrato de experiência	4	0,21
70	auxílio moradia	3	0,16	86	extinção do contrato	4	0,21
71	alimentação	3	0,16	87	causa do afastamento	3	0,16
72	recursos humanos	3	0,16				

Fonte: elaborada pelo autor.

Os termos listados pelas Figuras 27, 28, 29.1 e 29.2 foram selecionados arbitrariamente pelo autor deste projeto por pertencerem à temática proposta por cada categoria.

9.5.4 Implementação no sistema em C#

Após a etapa de seleção das palavras que seriam utilizadas, foi criado o modelo *bag-of-words* na classe Constantes.cs do sistema que contém algumas informações estáticas como endereços IP e caminhos para o servidor.

A implementação deste modelo foi realizada a partir de um vetor estático público do tipo *String* para cada uma das três categorias contendo todos os termos em seu respectivo setor, além de variações linguísticas destes termos.

Por último, um algoritmo de análise foi desenvolvido com o propósito de processar todo conteúdo textual que é recebido pelo sistema quando o usuário escolhe um arquivo para salvar no servidor. O resultado obtido é a classificação automática do documento em uma das três categorias (ou ainda em “Outros”, caso o arquivo não pertença a nenhuma das categorias anteriores).

9.6 ARMAZENAMENTO E ORGANIZAÇÃO DO CONHECIMENTO

Foi utilizado como meio de armazenamento de dados o banco de dados MySQL, pois se trata de uma ferramenta gratuita que é bastante utilizada em junção à linguagem C#. Em adição, o MySQL é uma plataforma muito frequentemente vista no contexto de desenvolvimento e sua popularidade é proporcional à segurança, eficiência e confiabilidade que a ferramenta pode proporcionar.

O banco de dados foi necessário com o propósito de armazenar, em adição às propriedades específicas do próprio sistema e sua estrutura, os dados em suas respectivas classes e as informações processadas pelo sistema em suas respectivas tabelas.

A seguir, são tratados os diagramas criados para representar a modelagem do banco de dados.

A Figura 30 contém a tabela que expressa os dados de login de um usuário. É importante observar que existe uma relação entre essa estrutura e a da tabela representada pela Figura 31, pois as ações que interagem com uma dessas estruturas também deve afetar a outra.

Figura 30 – Modelagem da tabela Users.

Coluna	Tipo	Constraint
user_id	integer	primary_key, autoincrement
user_username	varchar(20)	unique
user_password	varchar(30)	-
user_creation_date	timestamp	= current datetime
user_permission	varchar(1)	= default(1)

Fonte: elaborada pelo autor.

A Figura 31 contém a tabela que armazena as informações do usuário, além da ligação entre “Colaboradores” e “Users”, pois a coluna “col_user_id” é responsável por armazenar a relação entre as duas estruturas, tal que, para todos os efeitos, users.user_id == colaboradores.col_user_id.

Figura 31 – Modelagem da tabela Colaboradores.

Coluna	Tipo	Constraint
col_id	integer	primary_key, autoincrement
col_user_id	integer	foreign_key, = users.user_id
col_pnome	varchar(20)	-
col_snome	varchar(20)	-
col_cargo	varchar(30)	-

Fonte: elaborada pelo autor.

A Figura 32 demonstra a estrutura da tabela responsável pelo armazenamento das informações dos arquivos salvos no servidor. É importante ressaltar que esta tabela não contém o arquivo salvo no servidor, mas sim seus dados. Existe, no entanto, uma ligação direta entre o documento no servidor e a estrutura em questão, pois a coluna “caminhoArquivo” contém o caminho completo do arquivo armazenado no servidor.

Figura 32 – Modelagem da tabela Arquivos.

Coluna	Tipo	Constraint
idArquivo	integer	primary_key, autoincrement
nomeArquivo	varchar(30)	-
descArquivo	varchar(500)	-
dataUpload	timestamp	= current datetime
dataAlteracao	timestamp	= current datetime
idUserUpload	integer	-
idUserAlteracao	integer	-
caminhoArquivo	varchar(150)	-
tamanhoBytes	varchar(10)	-
keywordsArquivo	varchar(50)	-

Fonte: elaborada pelo autor.

9.7 TESTES DO SISTEMA

Com o propósito de testar a funcionalidade deste projeto, foi escolhido o método de caixa-branca, avaliando assim o processamento de documentos submetidos ao sistema.

A escolha dos textos foi dada a partir da seleção de 10 artigos, periódicos ou documentos escolhidos arbitrariamente pelo autor para representar cada categoria de setor implementada no sistema (comercial, financeiro e recursos humanos), totalizando 30 amostras distintas das produções textuais utilizadas durante a etapa de treinamento.

Após o período de seleção, o conteúdo textual foi inserido em documentos distintos rotulados e inseridos um a um no sistema comparando a saída obtida com a expectativa correspondente à temática de cada documento.

Em paralelo à avaliação da funcionalidade da ontologia, ainda foram conduzidos testes funcionais relacionados ao desempenho previsto para certos casos de uso do sistema. Para fins de avaliação, foram testados a integridade do conhecimento armazenado e o funcionamento dos aspectos de gerenciamento do projeto.

10 ELABORAÇÃO DO SOFTWARE

A elaboração do sistema no decorrer do desenvolvimento deste projeto pode ser vista como uma das etapas principais, pois se trata da aplicação dos estudos e do planejamento realizado até então. Os resultados são observados a partir das telas criadas para desempenhar cada função no escopo do sistema, conforme é abordado na próxima subseção.

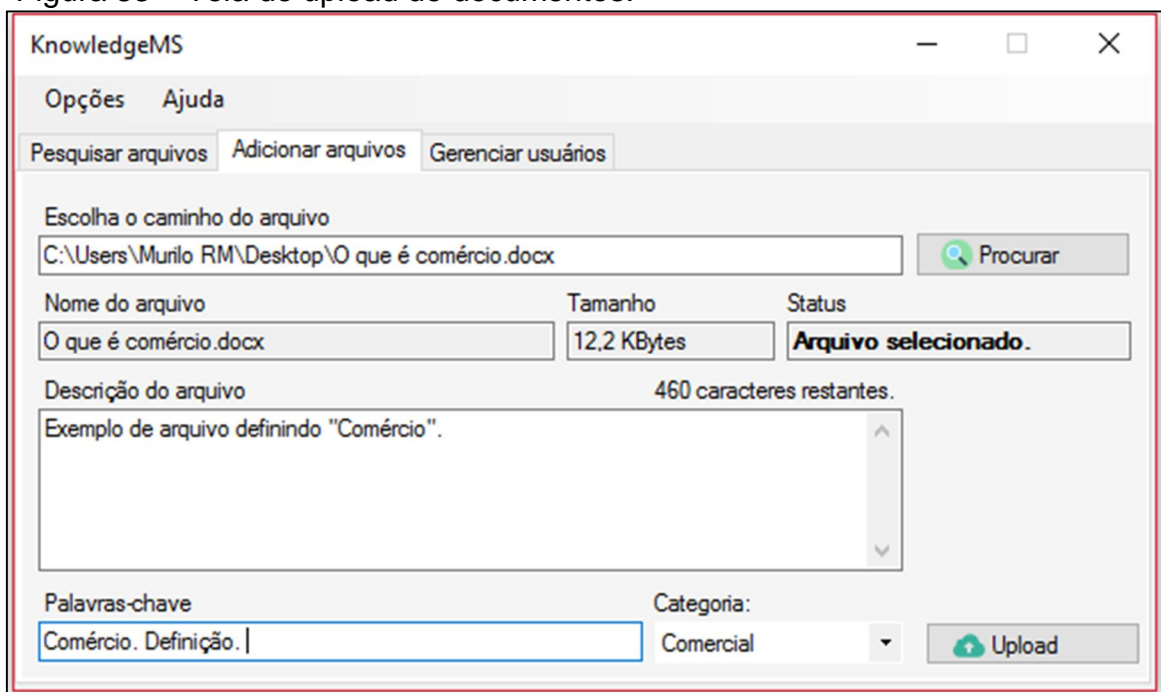
10.1 TELAS DO SISTEMAS

A seguir são abordadas as principais telas do sistema desenvolvido. Todas as demais telas podem ser encontradas na sessão APÊNDICE A – TELAS DO SISTEMA.

10.1.1 Upload de documentos

A Figura 33 trata do *upload* de documentos no sistema elaborado.

Figura 33 – Tela de upload de documentos.



The screenshot shows a web application window titled "KnowledgeMS" with standard window controls (minimize, maximize, close). The interface includes a menu bar with "Opções" and "Ajuda", and a navigation bar with "Pesquisar arquivos", "Adicionar arquivos", and "Gerenciar usuários". The main content area is for uploading a document. It features a text input field for the file path, currently containing "C:\Users\Murilo RM\Desktop\O que é comércio.docx", and a "Procurar" button. Below this is a table with three columns: "Nome do arquivo", "Tamanho", and "Status". The table contains one row with the file name "O que é comércio.docx", a size of "12.2 KBytes", and a status of "Arquivo selecionado.". Underneath the table is a text area for the "Descrição do arquivo" with a character count of "460 caracteres restantes." and the text "Exemplo de arquivo definindo 'Comércio'.". At the bottom, there is a "Palavras-chave" input field containing "Comércio. Definição.", a "Categoria:" dropdown menu set to "Comercial", and an "Upload" button with a green arrow icon.

Fonte: elaborada pelo autor.

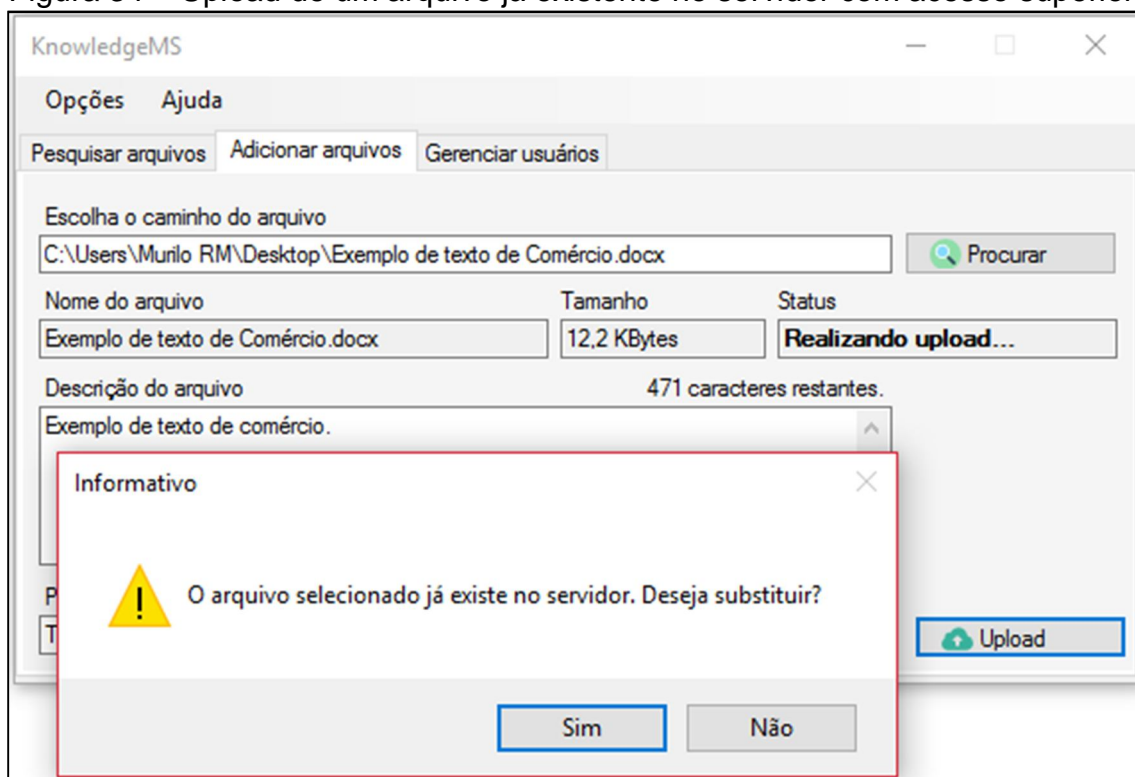
Conforme pode ser observado, o usuário inicia a interação selecionando o arquivo que será armazenado no servidor. Ao selecioná-lo, as informações “Nome do arquivo”, “Tamanho”, “Status” e “Categoria” são preenchidos automaticamente. A categoria do arquivo é dada pela classificação do mesmo de acordo com seu conteúdo partindo do ponto de vista da ontologia que foi previamente estabelecida. É importante ressaltar que entre as informações preenchidas automaticamente, apenas a categoria pode ser alterada.

A informação “Status” do arquivo é atualizada em tempo real no decorrer do processo de armazenamento do arquivo no servidor, e tem o propósito de informar o colaborador sobre o andamento de todo o procedimento.

A tela de *upload* ainda conta com informações opcionais que podem ser preenchidas pelo usuário para aumentar o nível de detalhes do arquivo armazenado no servidor. Essas informações são “**Descrição do arquivo**” – uma caixa de texto que permite ao colaborador detalhar as informações contidas no arquivo em até 500 caracteres – e “**Palavras-chave**” – uma caixa de texto que permite ao colaborador incluir palavras-chave sobre a temática do arquivo que está sendo armazenado no servidor.

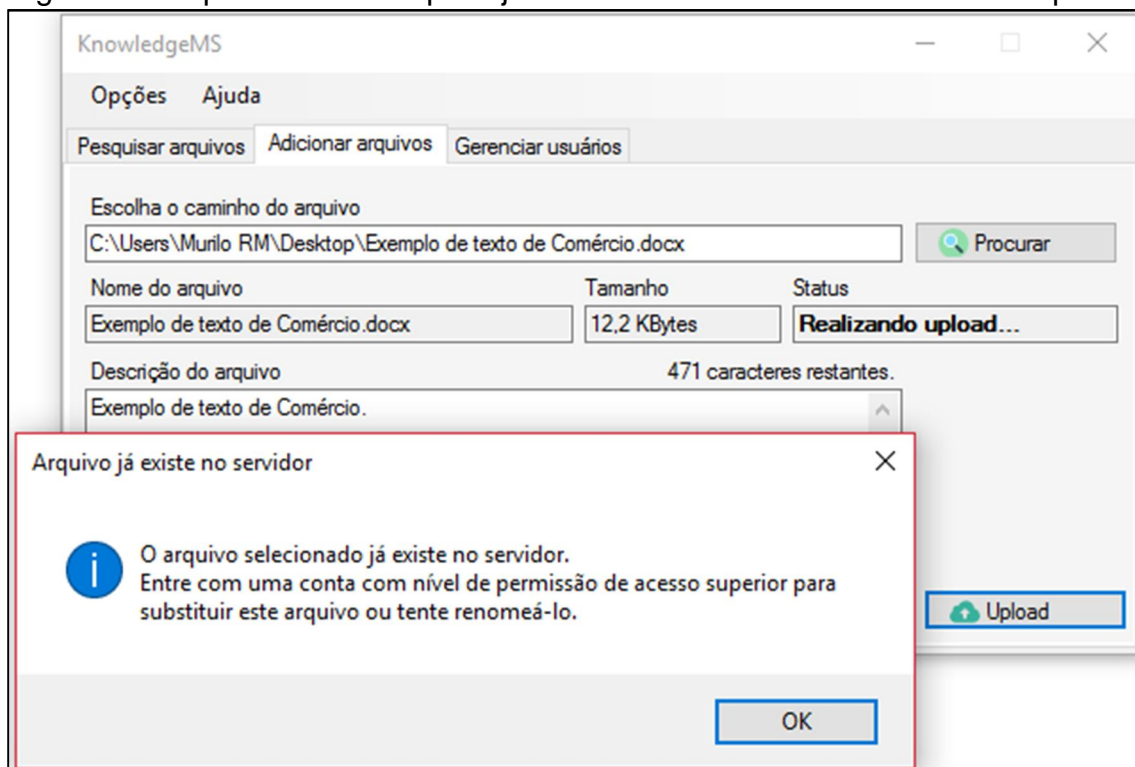
Em nível de acesso, é importante deixar claro que apenas um usuário com acesso superior pode sobrescrever um arquivo que já existe no servidor, enquanto um usuário com acesso regular é apenas avisado sobre o fato de que o arquivo que deseja salvar já existe. Essas interações podem ser vistas a seguir nas Figuras 34 e 35, respectivamente.

Figura 34 – Upload de um arquivo já existente no servidor com acesso superior.



Fonte: elaborada pelo autor.

Figura 35 – Upload de um arquivo já existente no servidor sem acesso superior.



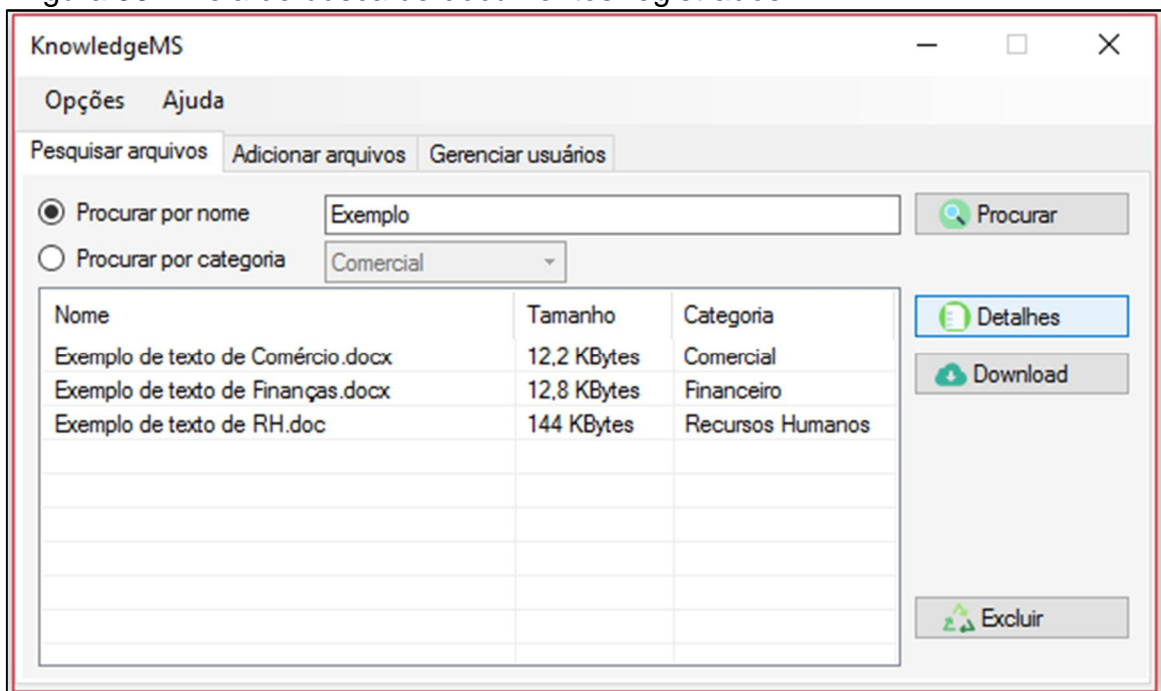
Fonte: elaborada pelo autor.

Essa diferença entre as duas possibilidades apresentadas pelas Figuras 34 e 35 é importante para fins de integridade do conhecimento armazenado no sistema.

10.1.2 Busca de documentos

A Figura 36 trata da tela de busca dos documentos que foram cadastrados no sistema e se encontram devidamente registrados no servidor.

Figura 36 – Tela de busca de documentos registrados.



Fonte: elaborado pelo autor.

Conforme pode ser observado, esta tela apresenta, a princípio, informações básicas sobre os documentos que foram cadastrados no sistema.

A busca por arquivos pode ser realizada de duas formas: **por nome** – na qual o sistema procurará pelo termo inserido pelo usuário nos campos de “Nome do arquivo”, “Descrição do arquivo” e “Palavras-chave” e em seguida listará os conteúdos encontrados – e **por categoria** – na qual o sistema listará todos os documentos registrados em uma determinada categoria.

A seguir, o colaborador pode selecionar um dos itens encontrados na busca para realizar as ações de visualizar detalhes sobre o arquivo, realizar *download* do documento pelo servidor ou excluí-lo do servidor.

A opção de visualizar detalhes, disponível através do botão “Detalhes” abre uma nova tela com informações mais específicas sobre o determinado arquivo, tal como sobre sua inserção no servidor, conforme pode ser visualizado na sessão APÊNDICE A – TELAS DO SISTEMA.

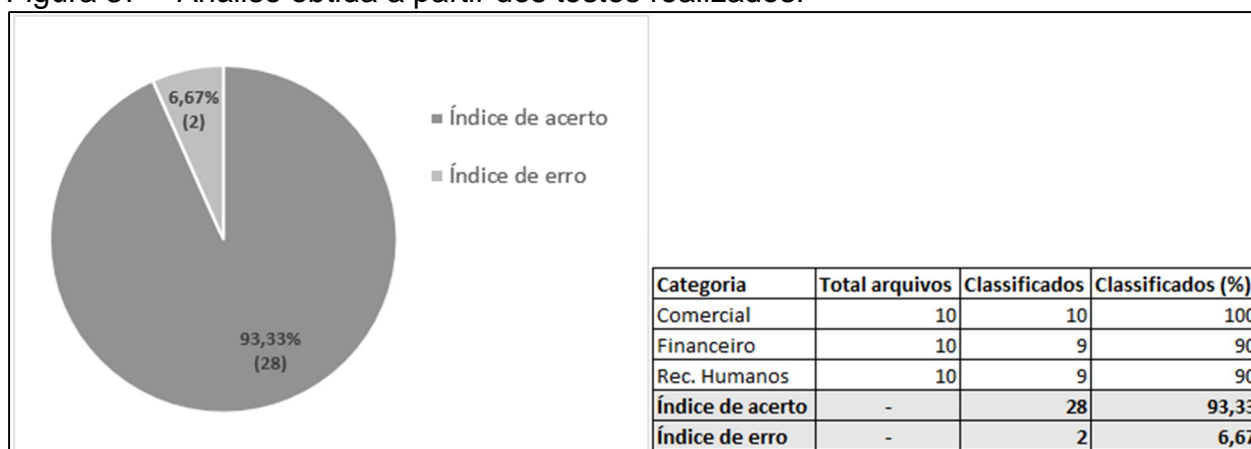
A opção de realizar *download* do documento abre uma tela com as informações do arquivo que será baixado. A partir de então, o usuário deve selecionar o local em seu computador para receber o documento e aguardar a finalização do *download*. Esta tela se encontra na sessão APÊNDICE A – TELAS DO SISTEMA.

A opção de excluir o arquivo do servidor só pode ser utilizada por um usuário com permissão superior no sistema. Caso contrário, o botão se encontrará desativado e não poderá ser clicado por um usuário comum, medida esta que foi tomada com o propósito de proteger a integridade dos arquivos contidos no sistema. Ao clicar nesta opção, o usuário visualizará as informações do arquivo e terá a oportunidade de confirmar a exclusão do arquivo ou cancelá-la ao fechar a janela que foi aberta. A ilustração correspondente a esta tela se encontra na sessão APÊNDICE A – TELAS DO SISTEMA.

11 VALIDAÇÃO DOS RESULTADOS OBTIDOS

Conforme tratado anteriormente, a etapa de testes foi realizada a partir da avaliação de resultados para 10 documentos por cada categoria. O resultado dessa análise pode ser observado a partir da Figura 37.

Figura 37 – Análise obtida a partir dos testes realizados.



Fonte: elaborada pelo autor.

A Figura 37 aborda o índice de acerto do algoritmo escolhido partindo de uma amostra de 30 documentos escolhidos arbitrariamente pelo autor. Conforme observado, os resultados indicam que foi alcançado um padrão satisfatório para a classificação de documentos, embora exista ainda espaço para otimização e aperfeiçoamento dos processos envolvidos no processamento de documentos pelo sistema.

É interessante de se ressaltar que a primeira etapa de testes envolveu apenas 15 documentos e uma taxa de erro de 2 documentos, totalizando um índice de acerto de 86,67%. A segunda etapa, portanto, adicionando 15 produções textuais ao teste inicial, aumentou em aproximadamente 9,24% o percentual de acerto do algoritmo de classificação, pois não houve demais erros no processamento do setor para cada documento avaliado pelo sistema.

12 TRABALHOS FUTUROS

O desenvolvimento deste projeto aplicado ao contexto empresarial deixou claro o espaço que há para evolução tanto da temática de gestão do conhecimento quanto do próprio sistema que foi elaborado em seus diversos aspectos.

Em nível da temática escolhida, é possível afirmar que o conceito por trás deste projeto poderia ser aplicado a qualquer meio em que existem grandes conjuntos de conhecimento sendo tratados de algum modo. Para tal, convém que exista uma forma de adicionar palavras (ou ainda extraí-las automaticamente) partindo do próprio usuário. Esta funcionalidade permitiria uma flexibilidade maior na adaptação do sistema ao vocabulário em um determinado contexto, assim como a melhoria na qualidade do vocabulário previamente existente e sua manutenibilidade.

A respeito das classes de palavras implementadas no sistema, também é possível notar a partir dos resultados obtidos que há espaço para otimização. Essas melhorias poderiam ser realizadas partindo-se do vocabulário desenvolvido para a ontologia ou para a forma como o algoritmo trata dos documentos processados.

Tratando-se de um sistema de gestão empresarial, nos dias de hoje é possível notar a partir de tendências de grandes serviços do ramo como SAP e Totvs que existe um movimento que tem transformado a plataforma de execução de produtos como este, deixando de lado cada vez mais as estações de trabalho com aplicações *desktop* aumentando a tecnologia e o investimento em plataformas *web* e *mobile*. Levando em consideração esta perspectiva, fica claro que uma evolução natural deste projeto envolveria meios de execução em sistemas que funcionam no próprio navegador e/ou em dispositivos portáteis.

Outro aspecto a ser tratado é a área de recuperação de informação que ocorre na etapa de elaboração da Ontologia e que poderia ser mais explorada a partir de estudos e aplicações complementares com o objetivo de enriquecer o esquema de processamento e seleção de documentos.

13 CONSIDERAÇÕES FINAIS

O projeto proposto no início da disciplina de Trabalho de Conclusão de Curso I evoluiu de acordo com diversos fatores no decorrer das etapas de pesquisa e desenvolvimento em virtude de problemas encontrados em relação à proposta inicial e a novas ideias que contribuíram com o aumento da qualidade do trabalho.

Ambos os fatores foram importantes para o período de desenvolvimento e colaboraram de forma a aumentar a complexidade do sistema. Os problemas encontrados no decorrer deste projeto em específico foram responsáveis, como exemplo, pela implementação do serviço de FTP no servidor em contrapartida à proposta inicial de realizar a transferência de arquivos através do protocolo HTTP POST.

Em relação à proposta, é possível afirmar a partir dos resultados obtidos que foi concluída, demonstrando o potencial da gestão do conhecimento através da unificação do conhecimento tratado em um determinado contexto.

Certamente ainda existem diversos aspectos a serem explorados partindo do que foi desenvolvido no decorrer deste trabalho, conforme foi tratado no tópico de Trabalhos futuros.

Por fim, cabe ressaltar que este trabalho é único em sua categoria e seu autor considera este fato como uma oportunidade para incentivar mais produções com essa temática.

REFERÊNCIAS

ALBUQUERQUE, A. B. et al. Ensino de Engenharia de Software: Desafios, Estratégias de Ensino e Lições Aprendidas. **Puc-rio.br**, c2009. Disponível em: <<http://goo.gl/UgsZ1i>>. Acesso em: 9 maio. 2016.

ALCADIPANI, R.; CAVALCANTI, M. F. R. Ontologias, Epistemologia e Estudos Organizacionais Críticos: a contribuição de Deleuze para uma crítica organizacional pós-estruturalista. **Anpad.org.br**, 2010. Disponível em: <<http://goo.gl/5P5aLk>>. Acesso em: 27 abr. 2016.

ALMEIDA, J. A. G. et al. Sistema de gestão do conhecimento para Rede Nacional de Bancos de Leite Humano. **Scielo.br**, 2005. Disponível em: <<http://goo.gl/fnOqBp>>. Acesso em: 16 maio. 2016.

AMBROSIO, A. P. L.; MORAIS, E. A. M. Ontologias: conceitos, usos, tipos, metodologias, ferramentas e linguagens. **Ufg.br**, 2007. Disponível em: <<http://goo.gl/3cksjF>>. Acesso em: 27 abr. 2016.

BALTZAN P.; PHILLIPS, A. **Sistemas de Informação**. Porto Alegre: AMGH Editora, 2012. Disponível em: <<https://goo.gl/oyEzIK>>. Acesso em: 24 abr. 2016.

BANCOS de dados relacionais. **Portal Educação**, 25 abr. 2013. Disponível em: <<http://goo.gl/tcJexl>>. Acesso em: 18 abr. 2016.

BARBOSA, S. D. J.; SILVA, B. S. **Interação Humano-Computador**. Rio de Janeiro: Elsevier, 2010.

BASTOS, G. G. et al. A Ontologia na ciência da informação. **Sbu.unicamp.br**, 2007?. Disponível em: <<http://143.106.108.14/seer/ojs/index.php/rbci/article/view/369/248>>. Acesso em: 28 abr. 2016.

BAWA, J. **Computador e Saúde**. Traduzido por Eduardo Farias. São Paulo: Summus, 1997.

BEZERRA, C. et al. Ontologia para Construção de Ambientes Interativos de Aprendizagem. **CEIE** – Comissão Especial de Informática na Educação, 2006. Disponível em: <<https://goo.gl/kJgPNI>>. Acesso em: 3 nov. 2016.

BEZERRA, E. **Princípios de análise e projeto de sistemas com UML**. 3. Ed. Rio de Janeiro: Elsevier, 2015.

BOTELHO, M. Aplicação de Ontologias na Organização de Conteúdos para Apoio a Equipes de Desenvolvimento de Software. **Btd.egc.ufsc.br**, 2015. Disponível em: <<http://btd.egc.ufsc.br/?p=2011>>. Acesso em: 16 maio. 2016.

BROOKSHEAR, J. G. **Ciência da Computação**: uma visão abrangente. Tradução Eduardo Kessler Piveta. 11. Ed. Porto Alegre: Bookman, 2013.

CAFÉ, L. M. A.; VITAL, L. P. Ontologias e taxonomias: diferenças. **Ufmg.br**, 2011. Disponível em: <<http://goo.gl/EzDJHz>>. Acesso em: 17 maio. 2016.

CAMPOS, M. L. de A.; GOMES, H. E. TAXONOMIA E CLASSIFICAÇÃO: a categorização como princípio. **Ufba.br**, 2007. Disponível em: <<http://www.enancib.ppgci.ufba.br/artigos/GT2--101.pdf>>. Acesso em: 3 nov. 2016.

CAMPOS, M. L. de A.; MARCONDES, C. H. ONTOLOGIA E WEB SEMÂNTICA: O ESPAÇO DA PESQUISA EM CIÊNCIA DA INFORMAÇÃO. **Ufba.br**, 2008. Disponível em: <<https://goo.gl/k5kfQn>>. Acesso em: 3 nov. 2016.

CAPURRO, R.; HJØRLAND, B. **O conceito de informação**. Perspectivas em Ciência da Informação, Belo Horizonte, v.12, n.1, 2007.

CARDOSO, O. N. P.; MACHADO, R. T. M. Gestão do conhecimento usando data mining: estudo de caso na Universidade Federal de Lavras. **SciELO.br**, c2008.

Disponível em: <<http://goo.gl/zgQqN4>>. Acesso em: 28 abr. 2016.

CARNEIRO, J. L. Introdução a bancos de dados. **Jlcarneiro.com**, c2004. Disponível em: <<http://goo.gl/PWgUrE>>. Acesso em: 27 abr. 2016.

CARVALHO, A. C. P. L. F. et al. **Inteligência Artificial**: uma abordagem de aprendizado de máquina. Rio de Janeiro: LTC, 2011.

CARVALHO, C. L.; SANTOS, F. C. Aplicação da Inteligência Artificial em Sistemas de Gerenciamento de Conteúdo. **Ufg.br**, 2008. Disponível em: <<http://goo.gl/X3lt1J>>.

Acesso em: 16 maio. 2016.

CORREA, J.; SPINOLA, M. de M. Adoção, seleção e implementação de um ERP livre. **scielo.br**, c2015. Disponível em: <<http://goo.gl/3sjwRq>>. Acesso em: 22 abr. 2016.

COSTA, D. J. de A. **A importância da Tecnologia da Informação no Auxílio à Administração**. 2008. 48f. Monografia de conclusão (Bacharelado em administração financeira) – Faculdade de Alagoas, Maceió, 2008.

DALL’OGLIO, P. **PHP: Programando com Orientação a Objetos**. 3. Ed. São Paulo: Novatec, 2015. Disponível em: <<https://goo.gl/HBg730>>. Acesso em: 24 maio. 2016.

DATE, C. J. **INTRODUÇÃO A SISTEMAS DE BANCOS DE DADOS**. 8. Ed. Rio de Janeiro: Elsevier, 2004.

ELMASRI, R.; NAVATHE, S. B. **Sistemas de banco de dados**. 6. Ed. São Paulo: Pearson Education, 2011.

ERTEL, W. **Introduction to Artificial Intelligence**. Traduzido por Nathanael Black. Nova York: Springer, 2011.

ÉVORA, Y. D. M.; GUIMARÃES, E. M. P., Sistema de informação: instrumento para tomada de decisão no exercício da gerência. **Scielo.br**, c2004. Disponível em: <<http://www.scielo.br/pdf/ci/v33n1/v33n1a09>>. Acesso em: 25 abr. 2016.

FARIA, L., Inteligência humana. Abordagens biológicas e cognitivas. **Scielo.br**, c2003. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-863X2003000200011>. Acesso em: 29 maio. 2016.

FEI-FEI, L. et al., Bag-of-words models. **NYU.org**, 2012. Disponível em: <http://cs.nyu.edu/~fergus/teaching/vision_2012/9_BoW.pdf>. Acesso em: 3 nov. 2016.

FERTIG, C.; MEDINA, M. **Algoritmos e Programação: Teoria e Prática**. São Paulo: Novatec, 2005.

FILHO, M. P. de C., Os sistemas de informação e as modernas tendências da tecnologia e dos negócios. **Scielo.br**, c1994. Disponível em: <<http://www.scielo.br/pdf/rae/v34n6/a05v34n6.pdf>>. Acesso em: 25 abr. 2016.

FOWLER, M., **UML Essencial: Um breve guia para a linguagem-padrão de modelagem de objetos**. Traduzido por João Tortello. 3. Ed. São Paulo: Bookman, 2007.

GONÇALVES, J. A.; SOUZA, R. R. Relações e conceitos em ontologias: teorias de Farradane e Dahlberg. **Uff.br**, 2007. Disponível em: <<http://www.uff.br/ontologia/artigos/15.pdf>>. Acesso em: 27 abr. 2016.

GONGORA, A. D. O que é Inteligência Artificial? **Ufsc.br**, 2007. Disponível em: <<http://goo.gl/ULu0Rh>>. Acesso em: 13 maio. 2016.

GONZALEZ, M.; LIMA, V. L. S. Recuperação de Informação e Processamento da Linguagem Natural. **PUCRS.br**, 2003. Disponível em: <<https://goo.gl/yJo8ON>>. Acesso em: 3 nov. 2016.

HOSSAIN, L.; PATRICK, J. D.; RASHID M. A., The Evolution of ERP Systems: A historical Perspective. **Faculty.biu.ac.il**, c2002. Disponível em: <<https://goo.gl/YeJoKg>>. Acesso em: 22 abr. 2016.

SANTIAGO JUNIOR, J. R. S. **Gestão do conhecimento**: A chave para o sucesso empresarial. São Paulo: Novatec, 2004.

KLAUSEN, P. Introduction to programming and the C# language. **Bookboon.com**, c2012. Disponível em: <<http://goo.gl/JAtf71>>. Acesso em: 21 abr. 2016.

LAUDON, J.; LAUDON, K., **Sistemas de Informação Gerenciais**. Traduzido por Luciana do Amaral Teixeira. 9. Ed. São Paulo: Pearson Education, 2011.

LIBERTY, J, **Programming C#**. 4. Ed. Sebastopol: O'Reilly, 2005. Disponível em: <<https://goo.gl/VcM8pt>>. Acesso em: 22 abr. 2016.

LIMA, V. L. S.; NUNES, M. G. V.; VIEIRA, R. Desafios do Processamento de Linguas Naturais. **Pucrs.br**, 2007. Disponível em: <<http://goo.gl/LsMQsv>>. Acesso em: 13 maio. 2016.

MARTINEZ, M. UML. **InfoEscola**, 2006-2016. Disponível em: <<http://www.infoescola.com/engenharia-de-software/uml/>>. Acesso em: 4 nov. 2016.

MARTINS, A. L. Potenciais aplicações da Inteligência Artificial na Ciência da Informação. **Uel.br**, 2010. Disponível em: <<http://goo.gl/mbDXmf>>. Acesso em: 13 maio. 2016.

Martins, C. A.; MATSUBARA, E. T.; MONARD, M. C., PreText: uma ferramenta para pré- processamento de textos utilizando a abordagem bag-of-words.

ResearchGate.Net, 2016. Disponível em: <<https://goo.gl/qzVMx3>>. Acesso em: 3 nov. 2016.

MATTAR, F. N. Pesquisa de Marketing, edição compacta. 5. Ed. Rio de Janeiro: Elsevier, 2012.

MELO, A. C., **Desenvolvendo Aplicações com UML 2.2**: do conceitual à implementação. 3. Ed. São Paulo: Brasport, 2011.

MICROSOFT DEVELOPER NETWORK. **MSDN**, c2016. Contém a documentação oficial da linguagem C#. Disponível em: <<https://goo.gl/tVfvpe>>. Acesso em: 21 abr. 2016.

MÖSSENBÖCK, H., Introduction to C#: The new language for Microsoft .NET. **uni-linz.ac.at**, c2001. Disponível em: <<http://goo.gl/g4Qj16>>. Acesso em: 25 abr. 2016.

NAKOV, S. et al. **Fundamentals of Computer Programming with C#**. Sofia: Faber, 2013.

NIEDERAUER, J. **Desenvolvendo Websites com PHP**. 2. Ed. São Paulo: Novatec, 2011.

NILSSON, N. J. **Principles of Artificial Intelligence**. Palo Alto: Morgan Kaufmann Publishers, 1980.

NONAKA, I.; TAKEUCHI, H., **Criação de conhecimento na empresa**: Como as empresas japonesas geram a dinâmica da inovação. Tradução por Oxford University Press. 20. Ed. Rio de Janeiro: Elsevier, 1997. Disponível em: <<https://goo.gl/CJKXBZ>>. Acesso em: 25 abr. 2016.

NOVO, H. F., A TAXONOMIA ENQUANTO ESTRUTURA CLASSIFICATÓRIA: UMA APLICAÇÃO EM DOMÍNIO DE CONHECIMENTO INTERDISCIPLINAR. **Ufba.br**, 2016. Disponível em: <<https://www.repositorio.ufba.br/ri/handle/ri/1202>>. Acesso em: 3 nov. 2016.

OLIVEIRA, C. H. P., **SQL: curso prático**. 9. Ed. São Paulo: Novatec, 2002.

O que há de mais humano na gestão. **Udesc.br**, c2006. Disponível em: <<http://goo.gl/oGaJBy>>. Acesso em: 25 abr. 2016.

PADILHA, T. C. C.; MARINS, F. A. S. Sistemas ERP: características, custos e tendências. **SciELO.br**, c2005. Disponível em: <<http://www.scielo.br/pdf/%0D/prod/v15n1/n1a08.pdf>>. Acesso em: 23 abr. 2016.

PEZZÈ, M.; YOUNG, M., **Teste e Análise de Software**: processos, princípios e técnicas. Traduzido por Bernardo Copstein e Flavio Moreira de Oliveira. Porto Alegre: Artmed, 2008.

PRESSMAN, R. S., **Engenharia de Software**: Uma abordagem profissional. 7. Ed. São Paulo: Artmed, 2011.

RASHEED, F., **Programmer's Heaven: C# School**. Fuengirola: NESTENIUS; WORTHINGTON; WRIGHT, 2006.

ROSA, J. L. G., **Fundamentos da Inteligência Artificial**. Rio de Janeiro: LTC, 2011.

ROSSETTI, A. G., Um modelo conceitual de gestão do conhecimento para unidades organizacionais de pesquisa agropecuária sob a ótica da interdisciplinaridade. **Embrapa.br**, c2009. Disponível em: <<http://goo.gl/n9ODwE>>. Acesso em: 25 abr. 2016.

SANVITO, W. L., Inteligência biológica versus Inteligência Artificial: uma abordagem crítica. **Scielo.br**, c1995. Disponível em: <www.scielo.br/pdf/anp/v53n3a/01.pdf>. Acesso em: 29 maio. 2016.

SHARP, J. **Microsoft® Visual C#® 2010: Step by Step**. New York: Microsoft Press, 2010.

SILVA, H. da, **TeO: um chatterbot para telessaúde**. 2014. 103f. Trabalho de conclusão de curso (Graduação em Ciência da Computação) – Universidade Sagrado Coração, Bauru, 2014.

SOMMERVILLE, I. **Engenharia de Software**. 9. Ed. São Paulo: Pearson Education, 2011.

SOUZA, C., Conceitos de Orientação a Objetos. **Ufpa.br**, 2016. Disponível em: <<http://www.ufpa.br/cdesouza/teaching/cedai/3-OO-concepts.pdf>>. Acessado em: 04 nov. 2016.

SUEHRING, S. **MySQL Bible**. Nova Jersey: Wiley Publishing, 2002.

WAZLAWICK, R. S. **Engenharia de software: conceitos e práticas**. Rio de Janeiro: Elsevier Editora, 2013.

WERNER, I. A.; WERNER, W. **Gestão do conhecimento: ferramentas tecnológicas e portais do conhecimento para empresas desenvolvedoras de tecnologia de médio e pequeno portes**. **Unifil.br**, c2004?. Disponível em: <<http://goo.gl/qUw5BW>>. Acesso em: 28 abr. 2016.

WIDENIUS, M. M.; AXMARK, D. **MySQL Reference Manual: Documentation from the source**. Distribuição online: O'Reilly, 2002. Disponível em: <<https://goo.gl/O97UNN>>. Acesso em: 15 maio. 2016.

ZAIDAN, F. A. **Gestão do Conhecimento na Engenharia de Software**. **Techoje.com.br**, c2010?. Disponível em: <<http://goo.gl/QGtTjp>>. Acesso em: 28 abr. 2016.

APÊNDICE A – TELAS DO SISTEMA

Este apêndice contém uma abordagem breve sobre as telas desenvolvidas para cobrir diversos aspectos do sistema. A seguir, essas telas são tratadas.

A tela de login do sistema, conforme pode ser visto pela Figura 38, contém os campos de “Login” e “Senha” para autenticação do usuário. Esta tela ainda é responsável por levar o usuário à tela de registro a partir do botão “Registrar”.

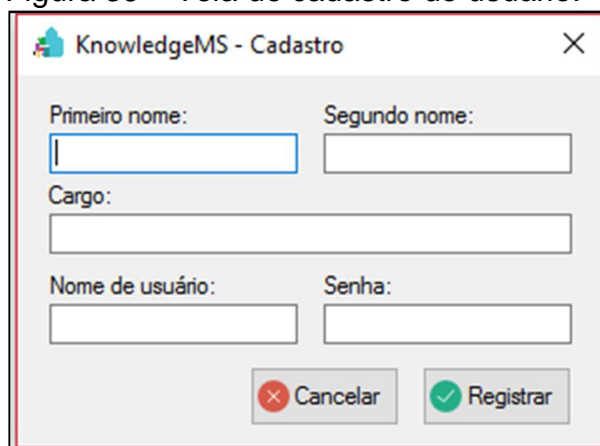
Figura 38 – Tela de login.



Fonte: elaborada pelo autor.

A Figura 39 ilustra a tela responsável pelo cadastro de um novo usuário. São gravadas algumas informações sobre este usuário para fins de referência e gerenciamento. É importante de se ressaltar que todo usuário cadastrado nesta tela possui por padrão nível de permissão “1”, o que o impede de excluir ou sobrescrever um documento, bem como, visualizar informações sobre outros usuários cadastrados no sistema.

Figura 39 – Tela de cadastro de usuário.



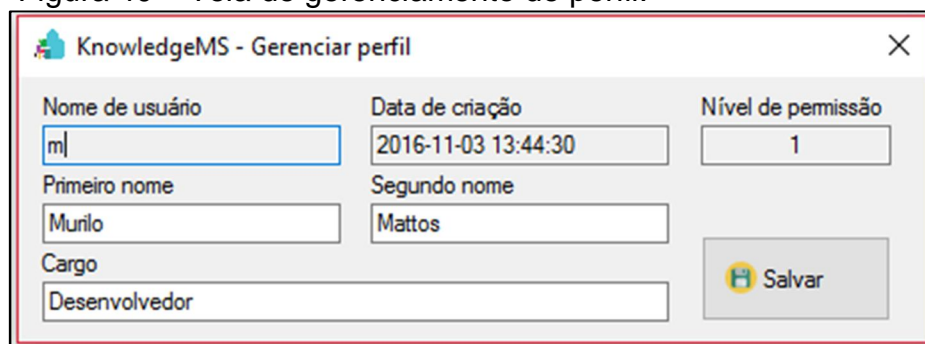
A janela de cadastro do KnowledgeMS contém os seguintes campos e botões:

Primeiro nome:	Segundo nome:
<input type="text"/>	<input type="text"/>
Cargo:	
<input type="text"/>	
Nome de usuário:	Senha:
<input type="text"/>	<input type="text"/>
<input type="button" value="Cancelar"/> <input type="button" value="Registrar"/>	

Fonte: elaborada pelo autor.

A tela representada pela Figura 40 corresponde ao serviço de gerenciamento do perfil de usuário. Embora todos os usuários possam ter acesso a esta sessão, nenhuma informação pertinente à estrutura e execução do sistema pode ser alterada a partir dela.

Figura 40 – Tela de gerenciamento de perfil.



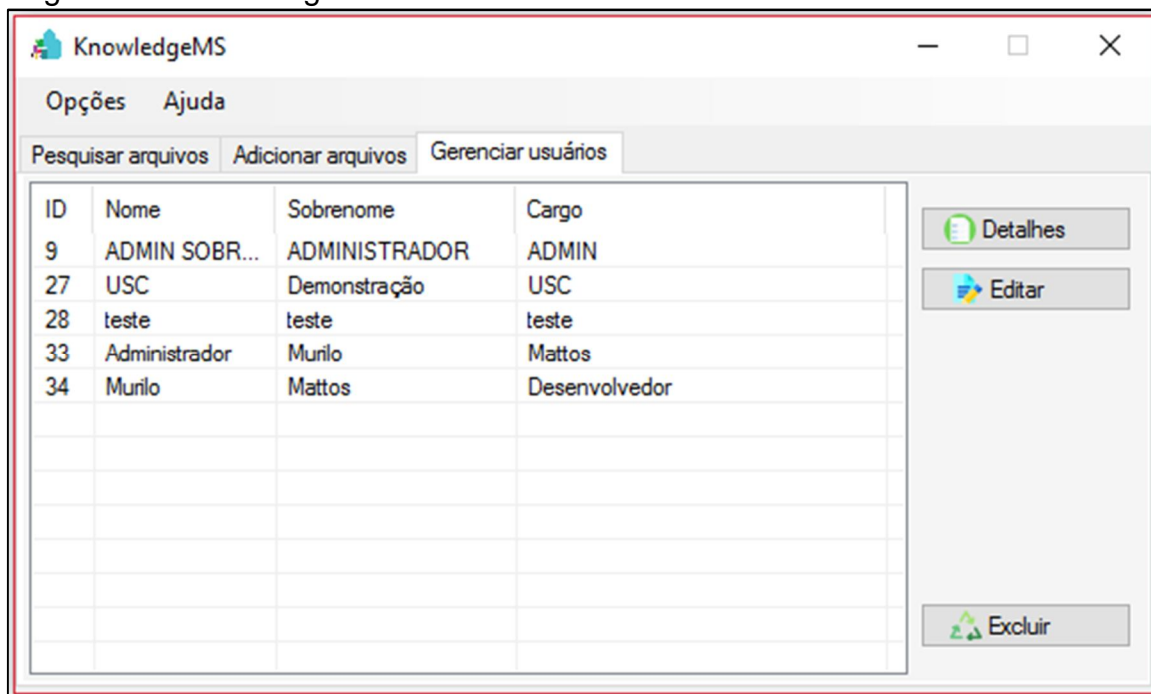
A janela de gerenciamento de perfil do KnowledgeMS contém os seguintes campos e botões:

Nome de usuário	Data de criação	Nível de permissão
<input type="text" value="m "/>	<input type="text" value="2016-11-03 13:44:30"/>	<input type="text" value="1"/>
Primeiro nome	Segundo nome	
<input type="text" value="Murilo"/>	<input type="text" value="Mattos"/>	
Cargo		<input type="button" value="Salvar"/>
<input type="text" value="Desenvolvedor"/>		

Fonte: elaborada pelo autor.

A Figura 41 demonstra a aba de gerenciamento de usuários que pode ser acessada somente por um usuário com permissão superior. Esta tela lista brevemente informações sobre todos os usuários cadastrados no sistema, assim como apresenta algumas funções que podem ser visualizadas ao selecionar um determinado item da lista.

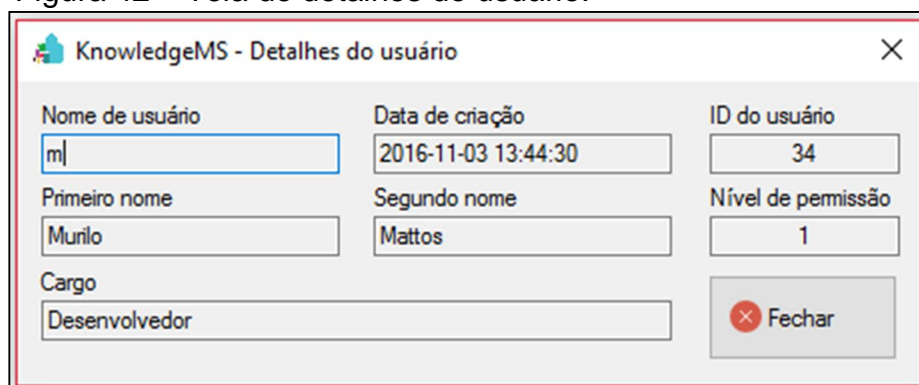
Figura 41 – Tela de gerenciamento de usuários.



Fonte: elaborada pelo autor.

Esta tela ilustrada pela Figura 42 contém informações detalhadas sobre o cadastro de um determinado usuário. É importante ressaltar, no entanto, que esta sessão existe apenas para fins de consulta, não sendo possível editar qualquer tipo de informação a partir dela.

Figura 42 – Tela de detalhes do usuário.



Fonte: elaborada pelo autor.

A Figura 43 trata da tela responsável por editar informações de um determinado colaborador. A partir desta sessão, é possível alterar o nível de permissão de um

usuário possibilitando-o a ter acesso privilegiado a informações e ações relacionadas ao sistema.

Figura 43 – Tela para edição de usuário.

Nome de usuário	Data de criação	ID do usuário
m	2016-11-03 13:44:30	34
Primeiro nome	Segundo nome	Nível de permissão
Murilo	Mattos	2
Cargo		
Desenvolvedor	Salvar	

Fonte: elaborada pelo autor.

A tela representada pela Figura 44 conta com informações detalhadas sobre o cadastro de um usuário que está prestes a ser excluído do sistema. A visualização desta tela, no entanto, não é o bastante para a exclusão definitiva de um colaborador, sendo assim necessário clicar em “Excluir” para confirmar a ação desejada.

Figura 44 – Tela de exclusão de usuário.

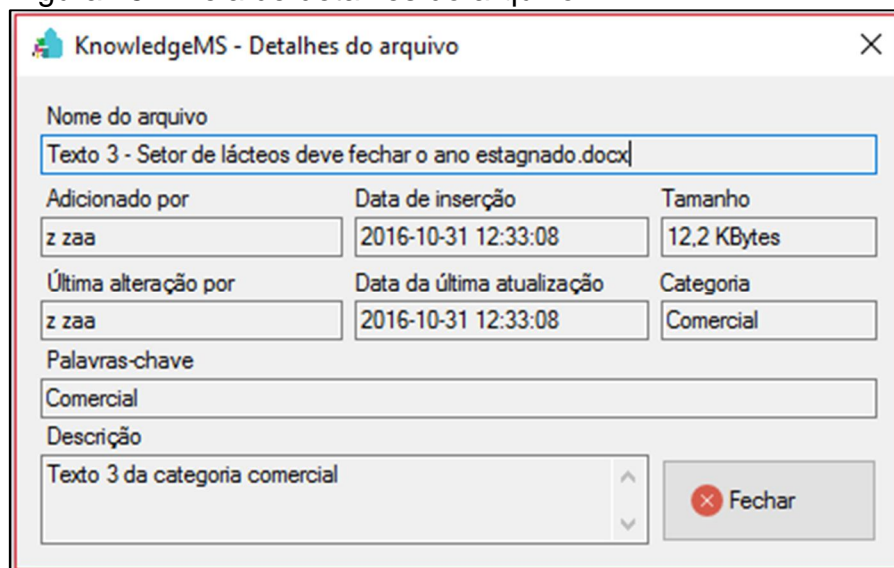
Nome de usuário	Data de criação	ID do usuário
m	2016-11-03 13:44:30	34
Primeiro nome	Segundo nome	Nível de permissão
Desenvolvedor	Murilo	2
Cargo		
Mattos	Excluir	

Fonte: elaborada pelo autor.

Conforme foi tratado na abordagem sobre as telas cruciais do sistema anteriormente pela Figura 27, é possível realizar algumas formas de busca para listar os arquivos que estão cadastrados no sistema. A Figura 45 apresenta os detalhes que podem ser obtidos ao selecionar um desses arquivos. É importante ressaltar que esta

tela meramente exibe os detalhes de um determinado documento, não sendo possível alterá-lo através desta sessão.

Figura 45 – Tela de detalhes do arquivo.



The screenshot shows a window titled "KnowledgeMS - Detalhes do arquivo" with a close button (X) in the top right corner. The window contains the following information:

- Nome do arquivo:** Texto 3 - Setor de lácteos deve fechar o ano estagnado.docx
- Adicionado por:** z zaa
- Data de inserção:** 2016-10-31 12:33:08
- Tamanho:** 12,2 KBytes
- Última alteração por:** z zaa
- Data da última atualização:** 2016-10-31 12:33:08
- Categoria:** Comercial
- Palavras-chave:** Comercial
- Descrição:** Texto 3 da categoria comercial

A "Fechar" button with a red X icon is located at the bottom right of the window.

Fonte: elaborada pelo autor.

Ao selecionar um determinado arquivo listado na busca, é possível realizar *download* do mesmo através da sessão representada pela Figura 46. Ao clicar em "Download", o usuário deve escolher um diretório no qual o arquivo será transferido a partir do servidor e aguardar a conclusão da transferência.

Figura 46 – Tela de download de arquivo.

KnowledgeMS - Download do arquivo

Nome do arquivo
 Texto 2 - RH E OS SEUS DIVERSOS SETORES.docx

Adicionado por z zaa	Data de inserção 2016-10-31 12:34:05	Tamanho 2,3 MBytes
Última alteração por z zaa	Data da última atualização 2016-10-31 12:34:05	Categoria RecursosHumanos

Palavras-chave
Comercial

Descrição
 Texto 2 da categoria rh

Download

Status: **Aguardando**

Fonte: elaborada pelo autor.

A tela representada pela Figura 47 apresenta a última opção disponível ao selecionar um arquivo listado na aba de busca. É importante ressaltar que apenas um usuário com permissão superior tem acesso a este recurso e que a visualização desta tela não garante a exclusão de um documento. Para tal, é necessário confirmar a ação clicando em “Excluir”.

Figura 47 – Tela de exclusão de arquivo.

KnowledgeMS - Excluir arquivo

Nome do arquivo
 Texto 2 - RH E OS SEUS DIVERSOS SETORES.docx

Adicionado por z zaa	Data de inserção 2016-10-31 12:34:05	Tamanho 2,3 MBytes
Última alteração por z zaa	Data da última atualização 2016-10-31 12:34:05	Categoria RecursosHumanos

Palavras-chave
Comercial

Descrição
 Texto 2 da categoria rh

Excluir

Fonte: elaborada pelo autor.