

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO

AUDITCO – AUDITORIA DE HARDWARE E SOFTWARES
DE TI EM REDES MICROSOFT

DIEGO ANTONIO PAMPLONA

BLUMENAU
2018

DIEGO ANTONIO PAMPLONA

AUDITCO – AUDITORIA DE HARDWARE E SOFTWARES

DE TI EM REDES MICROSOFT

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Sistemas de Informação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Prof. Roberto Heinzle, Dr. - Orientador

**BLUMENAU
2018**

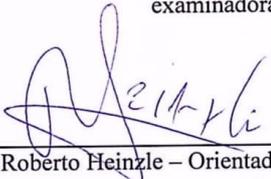
**AUDITCO: AUDITORIA E COMPLIANCE DE HARDWARE E
SOFTWARE PARA PEQUENAS E MEDIAS EMPRESAS**

Por

DIEGO ANTÔNIO PAMPLONA

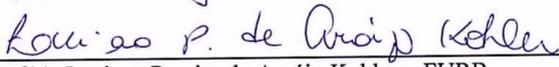
Trabalho de Conclusão de Curso aprovado
para obtenção dos créditos na disciplina de
Trabalho de Conclusão de Curso II pela banca
examinadora formada por:

Presidente:



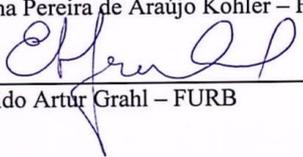
Prof(a). Roberto Heinzle – Orientador, FURB

Membro:



Prof(a). Luciana Pereira de Araújo Kohler – FURB

Membro:



Prof(a). Everaldo Artur Grahl – FURB

Blumenau, 12 de dezembro de 2018

Dedico este trabalho a minha esposa e parceira de todas as horas, Patrícia, aquela que está comigo em qualquer situação.

AGRADECIMENTOS

Agradeço aos meus pais que me trouxeram até a graduação não medindo esforços em me colocar nas melhores escolas possíveis e por me ensinarem a caminhar por conta própria, por acreditarem em mim e nunca demonstrarem dificuldades. Agradeço também a minha esposa Patrícia, que se não fosse por ela eu não teria chegado até aqui, por me guiar e segurar minha mão quando mais precisei e por entender minha ausência momentânea para desenvolver este trabalho. Por fim agradeço ao meu orientador que esteve comigo como mais do que um orientador, um amigo, e a Deus por me guiar e me proteger sempre.

Ninguém vai bater mais forte do que a vida.
Não importa como você bate e sim o quanto
aguenta apanhar e continuar lutando, o quanto
pode suportar e seguir em frente. É assim que
se ganha.

Sylvester Stallone – Rocky Balboa

RESUMO

Este trabalho apresenta o desenvolvimento de um software para auditoria de hardware e software para redes Microsoft. Usando a linguagem Visual Basic .NET e o framework .NET que facilitam o trabalho do desenvolvedor de software é apresentado um software que faz um levantamento de características físicas do computador auditado, como por exemplo processador, memória, disco, entre outras. Adicionalmente é feito o levantamento dos programas instalados, chaves de instalação do Windows e Office e por fim impressoras instaladas e ao final é gerado um arquivo no formato XML que é enviado para um servidor que processa e insere estes dados em uma base de dados relacional SQL Server.

Palavras-chave: SQL Server. .NET Framework. Visual Basic .NET. Licenciamento de software. Inventário de hardware.

ABSTRACT

This work presents the development of software for auditing hardware and software for Microsoft networks. Using the Visual Basic .NET language and the .NET framework that facilitate the work of the software developer is presented software that surveys the physical characteristics of the audited computer, such as processor, memory, disk, among others. Additionally, the installed programs are installed, Windows and Office installation keys and finally installed printers, and at the end a file is generated in XML format that is sent to a server that processes and inserts this data into a SQL relational database Server

Key-words: SQL Server. .NET Framework. Visual Basic .NET. Software licensing. Hardware inventory.

LISTA DE FIGURAS

Figura 1 - Tela Principal - Inventário Computadores Calesita.....	22
Figura 2 - Listagem do Inventário de Computadores	23
Figura 3 - Tela principal - lado cliente	25
Figura 4 - Relatório de Softwares Inventariados	26
Figura 5 - Atividades Executadas pelo Agente Cliente.....	29
Figura 6 - Atividades executadas pelo Agente Servidor	32
Figura 7 - Diagrama de Casos de Uso	33
Figura 8 - Principais classes do auditCo - Client	34
Figura 9 - Esquema de funcionamento entre o guest e o servidor.....	40
Figura 10 - Serviço AuditCO - Cliente instalado e pronto para o uso.....	40
Figura 11 - Serviço AuditCO - Server instalado e pronto para o uso.....	40
Figura 12 - AuditCo Server - tela de login	41
Figura 13 - AuditCo server - tela principal	41
Figura 14 - Tela de cadastro de usuários	42
Figura 15 - Tela de importação manual de XML	43
Figura 16 - Tela de Cadastro de Informações Adicionais	44

LISTA DE QUADROS

Quadro 1 - Listagem dos Programas Instalados	20
Quadro 2 - Ajuda Geral - Comandos Disponíveis.....	20
Quadro 3 - Ajuda - Módulo Grupos	21
Quadro 4 – Procedimento que lê parâmetros através de arquivo INI.....	35
Quadro 5 - Trecho do Código OnStart - Serviço Cliente	36
Quadro 6 - Parte do código da função CriaXML	37
Quadro 7 - Parte do XML gerado após uma coleta	38
Quadro 8 - Função Gravalog	38
Quadro 9 - Função de envio de e-mail	39
Quadro 10 - Comparativo de funcionalidades com os softwares correlatos	45
Quadro 11 – Conteúdo de XML gerado pelo software cliente.....	51

LISTA DE TABELAS

Tabela 1 - Cenários de teste.....	46
-----------------------------------	----

LISTA DE ABREVIATURAS E SIGLAS

ABES – Associação Brasileira de Empresas de Software

COBIT – Control Objectives for information and Related Technology

DNS – Domain Name System – Sistema de Nomes de Dominio

FTP – File Transfer Protocol – Protocolo de Transferência de Arquivos.

IOT – Internet Of Things – Internet das Coisas

IOT – Internet Of Things – Internet das Coisas

ISACA – Information Systems Audit and Control Association

ITAM – IT Aset Management – Gerenciamento de Ativos de TI

ITIL – Information Technology Infrastructure Library.

LAN – Local Area Network

PIB – Produto Interno Bruto

TCC – Trabalho de Conclusão de Curso

TCP – Transmission Control Protocol

TI – Tecnologia da Informação

USB – Universal Serial Bus

WAN - Wide Area Network

XML – Extensible Markup Language

SUMÁRIO

1 INTRODUÇÃO.....	13
1.1 OBJETIVOS.....	14
1.2 ESTRUTURA DO TRABALHO	15
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 REGISTRO DO WINDOWS	16
2.2 CONTROLE DE ATIVOS DE TI.....	16
2.3 PIRATARIA DE SOFTWARE.....	17
2.4 AUDITORIA DE SOFTWARE	17
2.5 COBIT	18
2.6 .NET FRAMEWORK	18
2.7 TRABALHOS CORRELATOS.....	19
2.7.1 Sistema de webservices para inventário de estações em rede.....	19
2.7.2 Sistema para inventario de computadores de redes Windows baseado na análise de registro na empresa Calesita.....	21
2.7.3 Software de inventário de software de equipamentos de rede utilizando session message block.....	24
3 DESENVOLVIMENTO DO SOFTWARE	27
3.1 LEVANTAMENTO DE INFORMAÇÕES	27
3.2 ESPECIFICAÇÃO	28
3.2.1 Diagrama de Atividades.....	28
3.2.2 Diagrama de casos de uso	32
3.2.3 Diagrama de Classes	34
3.3 IMPLEMENTAÇÃO	34
3.3.1 Técnicas e ferramentas utilizadas.....	35
3.3.2 Operacionalidade da implementação	39
3.4 RESULTADOS E DISCUSSÕES.....	44
4 CONCLUSÕES.....	47
4.1 EXTENSÕES	47
REFERÊNCIAS	49
ANEXO A – EXEMPLO DE XM GERADO PELO SOFTWARE CLIENTE	51

1 INTRODUÇÃO

Não é novidade que as auditorias nas empresas são uma rotina, sendo que todos os dias acontecem validações, como por exemplo, se os prazos estão sendo cumpridos corretamente, processos, metodologias, etc. Algumas auditorias são rotineiras, já outras acontecem de forma aleatória, enquanto umas são efetuadas pela própria empresa no papel de um auditor em um departamento de Gestão da Qualidade, outras são executadas por terceiros, com um objetivo específico. A Tecnologia da Informação (TI) é um mercado em crescimento contínuo, o qual possui participação de 7,1% do Produto Interno Bruto (PIB) do Brasil, por esse motivo abre espaço para surgir novas empresas produtoras de softwares para os mais diversos fins, algumas destas empresas já detentoras de outros produtos e serviços no mercado. Desta forma surgiu a possibilidade de lucrar com a produção do software e ao mesmo tempo com auditorias de uso destes nas empresas-cliente (VALOR ECONÔMICO, 2017, p.1).

Uma das questões principais, atualmente discutidas no âmbito jurídico das empresas, está ligada à Tecnologia da Informação (“TI”), mais precisamente às relações jurídicas que envolvem auditoria de softwares.

Com a evolução da tecnologia da informação e a competitividade do mundo atual, se torna cada vez mais necessário que uma empresa mantenha os procedimentos eficientes de produção e gestão, o que só é possível com a utilização adequada de *softwares* (SARTORE, 2014, p.1)

Nos últimos meses empresas como Microsoft, Autodesk e Adobe, produtoras, desenvolvedoras e fornecedoras de software, intensificaram ações de fiscalização e notificação contra o uso de programas irregulares ou piratas, principalmente em empresas de pequeno porte (HF TECNOLOGIA, 2018, p, 1).

No primeiro meio, os fabricantes analisam histórico de compras de computadores, notebooks e até softwares de servidores, se a empresa adquiriu nos últimos anos computadores sem sistema operacional e não há compras avulsas do sistema, é possível que ela esteja utilizando softwares irregulares, se a empresa adquiriu recentemente um Windows Server 2016 Standard mas não comprou as chamadas CALs de acesso, é possível que ela esteja irregular. Outro exemplo é pela quantidade de funcionários, um escritório de arquitetura com 05 funcionários e sem licenças de AutoCAD em seu nome, é muito provável que tenha algum computador com software irregular tornando-a alvo para fiscalização (HF TECNOLOGIA, 2018, p, 1).

Os fabricantes utilizam basicamente dois caminhos para chegar a empresas irregulares, o primeiro é através de pesquisa e levantamento de dados referente a faturamento, quantidade de funcionários, compras de equipamentos e softwares entre outros. O segundo é via denúncia.

A questão licenciamento de softwares passa por mudanças de forma gradativa, tendo inclusive nos últimos dois anos acontecido mudanças significativas no processo de licenciamento Microsoft. Essas mudanças surgem na onda de soluções virtualizadas ou em nuvem (RASMUSSEN, 2018, p.1). Baseado nesse cenário Rasmussen (2018) define que:

Partir para o uso de software pirata pode parecer algo tentador, mas, é necessário resistir a isso. Afinal, tal ação colocará a empresa em sérios riscos que podem comprometer os seus resultados.

Com a popularização da internet e a melhora na velocidade das conexões, tornaram-se comuns os métodos para destravar softwares pagos e contar com suas funcionalidades completas.

À primeira vista, tal cenário pode parecer bastante vantajoso para quem não quer gastar ou não dispõe de um grande orçamento. Todavia, a integridade dos arquivos digitais da sua empresa ficará comprometida com o uso de software pirata. (RASMUSSEN, 2018, p.1).

Nos últimos anos as empresas fabricantes de softwares, como a Microsoft, vêm ampliando canais de fiscalização de empresas quanto aos seus licenciamentos de softwares, com o objetivo de combater a pirataria (HF TECNOLOGIA, 2018, p. 1).

Esses canais incluem parceiros que enviam e-mails, ligações ou notificações, para empresas de pequeno e médio porte em todo Brasil, solicitando que o responsável pela empresa ou área de TI apresente levantamento dos equipamentos existentes na empresa, bem como das licenças de software, ou ainda, pedem que seja feita a instalação de software que realiza tal levantamento de forma automática (HF TECNOLOGIA, 2018, p. 1).

Baseado na argumentação apresentada, é alvo deste estudo o desenvolvimento de um software que abrange os levantamentos necessários para se obter uma gestão de softwares instalados em *notebooks* e *desktops* de uma empresa, de tal forma que o automatismo nos levantamentos feitos permita rapidez e fidelidade das informações obtidas.

Também espera-se que estas as informações sejam facilmente transmitidas a um agente intermediário que tratará a informação recebida e as persista em uma base de dados relacional, deixando disponíveis dados suficientes para uma tomada de decisão precisa e segura.

1.1 OBJETIVOS

O objetivo principal é desenvolver um software de levantamento de informações de hardware e software para controle e auditoria dos computadores de uma empresa.

Os objetivos específicos são:

- a) agrupar as informações levantadas de forma automatizada originadas dos clientes do software desenvolvido;
- b) Capturar e centralizar um controle de versão e licença dos softwares instalados;
- c) criar uma biblioteca de funções para levantamento de informações de hardware e software para softwares de auditoria em redes Microsoft;

1.2 ESTRUTURA DO TRABALHO

No primeiro capítulo tem-se a introdução ao tema deste trabalho com as justificativas e objetivos apresentados.

No segundo capítulo se apresenta a fundamentação teórica obtida através da pesquisa sobre os assuntos que norteiam a auditoria de software, como a definição de um programa de computador, registro do Windows, controle de ativos de TI, pirataria de software, auditoria de software, COBIT, .NET framework, bem como os trabalhos correlatos.

O terceiro capítulo apresenta o desenvolvimento do software iniciando desde o levantamento de informações e na sequência a especificação, implementação e, ao final, resultados e discussões.

No quarto capítulo são expostos as conclusões deste trabalho bem como se apresentadas sugestões para futuros trabalhos.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo será discorrido sobre os conceitos, técnicas e ferramentas que direcionam o trabalho e embasam a pesquisa feita. Serão mostrados assuntos necessários para compreensão deste trabalho, tais como registro do Windows, controle de ativos de TI, pirataria de software, auditoria de software, COBIT, .NET framework e os trabalhos correlatos.

2.1 REGISTRO DO WINDOWS

O Registro do Windows pode ser consultado através do comando regedit executado a partir da lupa de comandos ou do botão Windows / Iniciar ou da tecla de atalho Ctrl + esc. O Registro é demonstrado em formato de árvore de diretórios, sendo que é possível navegar pelos níveis até chegar na localização em que pode ser consultada e alterada as chaves já existentes ou criar novas chaves com valores.

O Registro é um local centralizado onde é possível manter as informações referentes a programas e periféricos instalados, como, também preferencias pessoais de usuário. A partir do Windows 95 a Microsoft vem orientando os desenvolvedores de sistemas na utilização do Registro (um único repositório) para armazenar as informações de seus programas e, assim de certa forma “desencorajando” o uso dos INI's que eram largamente utilizados nas aplicações desenvolvidas na plataforma Windows 3 (COSTA, RIBEIRO, 2004, p. 2).

De acordo com COSTA, RIBEIRO (2004) falando de forma simplificada, o registro é como um guia que mostra ao seu computador quais são os parâmetros a serem adotados no seu carregamento para a memória e durante o seu uso independentemente de quem a solicita (usuário, software ou hardware).

2.2 CONTROLE DE ATIVOS DE TI

Um inventário de TI bem executado e mantido atualizado pode auxiliar o gerente de tecnologia a ter todos os seus recursos de TI em pleno funcionamento e evitar qualquer tipo de surpresa desagradável. Sobre inventário TÉCNICA (2012) estabelece que um inventário atualizado de todos os Ativos de uma empresa é exigido para adequação às normas fiscais e contábeis, como a ABNT NBR ISO/IEC 17799, 2005 item 7.11 (MILVUS, 2018, p. 1)

O inventário dos ativos deve ser realizado periodicamente, uma tarefa não muito simples considerando o grande volume de ativos que uma empresa pode ter. Os ativos de TI são o coração de grande parte das empresas e maximizar este investimento requer um Sistema de Gestão de Ativos de TI - ITAM (IT Asset Management). Um ITAM é o conjunto de práticas de negócios que unem funções financeiras, contratuais e de inventário para apoiar a gestão do ciclo de vida e tomada de decisões estratégicas para o ambiente de TI. Os ativos incluem todos os

elementos de software e hardware que são encontrados no ambiente de negócios (MILVUS, 2018, p. 1)

As empresas de hoje buscam encontrar o equilíbrio entre o custo e a receita, e para isso tem se buscado modelos de gestão que FREITAS (2010) define que um modelo ótimo de empresa é aquele que possui um planejamento estratégico bem definido.

O Modelo de empresa que traduzido corretamente pelos gestores em processos empresariais executados pela área operacional produzirão bens e serviços com qualidade. Desta forma podemos considerar que os processos que norteiam o gerenciamento de ativos são tão importantes quando aqueles que geram o principal lucro da empresa.

2.3 PIRATARIA DE SOFTWARE

De acordo com a Associação Brasileira das Empresas de Software (ABES) pirataria é a prática de reproduzir ilegalmente um programa de computador, sem a autorização expressa do titular da obra e, conseqüentemente, sem a devida licença de uso. Conforme ORRICO JR (2004) o conceito de pirataria está originalmente relacionado à pilhagem, isto é, ao furto ou roubo do resultado do trabalho ou da propriedade de alguém por outrem.

Em suas considerações gerais ORRICO JR (2004) estabelece que:

No Brasil a *pirataria* de software teve início com a criação dos microcomputadores, na década de setenta do século XX, tendo se expandido fortemente em decorrência da criação, por lei da chamada “reserva de mercado”, cujo escopo era proteger a empresa e o mercado nacional.

De acordo com PORTAL ABES (2018) a Lei 9609/98 estabelece que a violação de direitos autorais de programas de computador é crime, punível com pena de detenção de seis meses a quatro anos e multa, além de ser passível de ação cível indenizatória.

2.4 AUDITORIA DE SOFTWARE

Com a evolução da tecnologia da informação e a competitividade do mundo atual, torna-se cada vez mais necessário que uma empresa mantenha os procedimentos eficientes de produção e gestão, o que só é possível com a utilização adequada de softwares (SARTORE, 2014, p. 1). Vê-se empresas com um inventário de cinco computadores recebendo notificações com o propósito de auditar o uso de software, onde o mais preocupante é que a maioria não tem ideia sobre como proceder em casos como esse (HF TECNOLOGIA, 2018, p. 1).

Quando uma empresa é pequena, como por exemplo uma contabilidade, é mais simples de controlar aquilo que está instalado em cada *desktop* ou *notebook* de cada

colaborador. Porém, quando se passa para um nível superior de empresa, e pode ser não tão grande assim, como nos casos de malharias, que trabalham com instalações físicas distantes e com a estrutura de TI descentralizadas é perceptível uma complexidade maior no controle dos artefatos de TI e dos softwares instalados nos *desktops* e *notebooks*. Sobre isso SARTORE (2014) afirma que se observa um aumento de usuários e muitos exemplos de licenças desatualizadas. Não por outro motivo, as empresas de software estão promovendo cada vez mais auditorias (SARTORE, 2014, p. 1).

Uma auditoria, além de ser o único modo seguro de determinar a extensão das violações de direitos, traz também muitos benefícios adicionais, entre eles, o de reforçar o controle da aplicação, da versão e a segurança dos dados, determinando o grau de exposição a riscos e a prevenção de desastres, sem contar o indesejado litígio judicial, ou seja, as auditorias de software são um benefício não só para o produtor do software, como também para o cliente. (SARTORE, 2014, p. 1).

Diante do exposto se vê como benefício para cliente e fornecedor que aconteçam as auditorias e desta forma automatizá-las é um ganho de produtividade diante da complexidade que pode se tornar o processo.

2.5 COBIT

A sigla significa Controle de Objetivos para Informação e Tecnologia Relacionada. É a base de conhecimento mais reconhecida e utilizada no mercado para apoiar organizações na Governança de Tecnologia da Informação (TI) (PORTAL GSTI, 2018, p. 1).

O framework é mantido pela Information Systems Audit and Control Association (ISACA) e descreve como gerenciar e controlar cada um dos componentes que pertencem à Governança Corporativa de Tecnologia da Informação. Os componentes são elementos denominados de habilitadores pelo COBIT. Os habilitadores são divididos nas seguintes categorias: processos, práticas, modelos, princípios, pessoas, informações, infraestrutura, aplicativos, estruturas organizacionais. Para cada Habilitador, o framework apresenta guias de boas práticas do como utilizá-lo para garantir o sucesso do seu sistema interno de Governança Corporativa de TI (PORTAL GSTI, 2018, p. 1).

O Framework evoluiu e hoje está na versão 5. Um dos grandes diferenciais da versão atual é a introdução do conceito de cascata de objetivos que auxilia as organizações na definição de objetivos e requisitos do negócio e ações para tecnologia da informação (PORTAL GSTI, 2018, p. 1).

2.6 .NET FRAMEWORK

O .NET (lê-se dotnet) *framework* é uma biblioteca de códigos da Microsoft o qual é possível desenvolver os mais diversos tipos de softwares desde os softwares cliente-servidor até softwares mais avançados como os de Internet Of Things (IOT) e *Cloud-Based* nuvem

Azure. O .NET é uma iniciativa da Microsoft que visa reunir em uma única plataforma as linguagens C#, F# e Visual Basic interligado com a biblioteca .NET. Com o .NET é possível criar aplicações Desktop, WEB, IOT, Jogos e Cloud-Based, tudo construído sobre a biblioteca .NET. Com o .NET pode-se usar diversos IDEs sendo de acordo com a Microsoft, o mais conhecido e usado o Visual Studio, uma poderosa IDE de desenvolvimento que apoia a construção de um software com ferramentas de teste do software, medições de consumo e autocritica do trecho de código durante a codificação (MICROSOFT DOCUMENTS, 2018, p.1).

Com o .NET framework é possível ter acesso a diversas bibliotecas, sejam elas nativas ou construídas por outros desenvolvedores. Através do Visual Studio é possível codificar o software desejado já embarcando o acesso às diversas bibliotecas disponíveis (MICROSOFT DOCUMENTS, 2018, p.1).

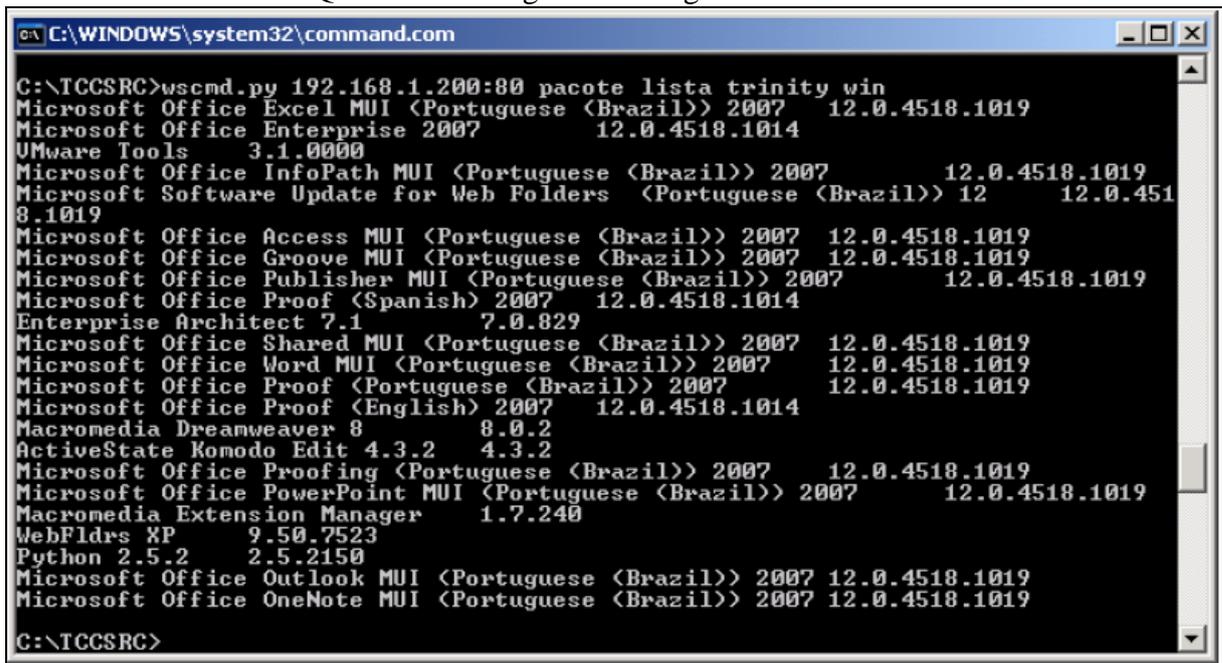
2.7 TRABALHOS CORRELATOS

Nesta seção serão apresentados três trabalhos correlatos com similaridades aos principais objetivos. O primeiro é um conjunto de *webservices* desenvolvidos para apoio à coleta de informações de uma rede heterogênea de computadores, um TCC de Babinetti (2008). O segundo é um TCC de Heiden (2013) que apresenta um software de inventário de redes Windows o qual foi alvo uma empresa de brinquedos. Por fim, o TCC de Kreuch (2007) que é um software de inventário de software, desenvolvido em continuidade a outro já existente, um software de levantamento patrimonial do parque de hardware.

2.7.1 Sistema de webservices para inventário de estações em rede

O trabalho é apresentado através do desenvolvimento de um software que possibilita o inventário de estações em uma rede heterogênea utilizando a tecnologia de webservices para coletar, processar, persistir e disponibilizar informações sobre as estações da rede. Dentre as principais funcionalidades estão o levantamento de processos em execução no momento da coleta e o diagnóstico de alterações/tarefas antes da transmissão dos dados ao centralizador das coletas. Outra funcionalidade é a listagem dos programas instalados de forma ordenada e precisa, sem qualquer perda de informações (BAMBINETTI, 2008).

Quadro 1 - Listagem dos Programas Instalados



```

C:\WINDOWS\system32\command.com
C:\TCGSRC>wscmd.py 192.168.1.200:80 pacote lista trinity win
Microsoft Office Excel MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office Enterprise 2007 12.0.4518.1014
UMware Tools 3.1.0000
Microsoft Office InfoPath MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Software Update for Web Folders <Portuguese <Brazil>> 12 12.0.451
8.1019
Microsoft Office Access MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office Groove MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office Publisher MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office Proof <Spanish> 2007 12.0.4518.1014
Enterprise Architect 7.1 7.0.829
Microsoft Office Shared MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office Word MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office Proof <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office Proof <English> 2007 12.0.4518.1014
Macromedia Dreamweaver 8 8.0.2
ActiveState Komodo Edit 4.3.2 4.3.2
Microsoft Office Proofing <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office PowerPoint MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Macromedia Extension Manager 1.7.240
WebFldrs XP 9.50.7523
Python 2.5.2 2.5.2150
Microsoft Office Outlook MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
Microsoft Office OneNote MUI <Portuguese <Brazil>> 2007 12.0.4518.1019
C:\TCGSRC>

```

Fonte: Bambinetti (2008).

Conforme figura 1, o ponto forte da aplicação é a flexibilidade de funcionamento desta em qualquer rede seja ela interna ou externa. A chamada da aplicação é feita no formato de acesso a qualquer site, ou seja, via Domain Name System (DNS) e por isso pode ser alterado o servidor de execução sem qualquer perda de informações, bem como o servidor de coleta pode funcionar local ou remoto sem qualquer ônus à aplicação cliente. O software ainda contém relatórios padrões para obtenção de listagem dos dados coletados e todos os dados são persistidos em base de dados relacional ficando sempre disponíveis para serem gerados novos relatórios com leiaute a escolha do administrador do sistema. Se pode destacar que é dotada da capacidade de gerar alertas caso haja alguma alteração no comportamento padrão da solução. Além disso, caso o usuário tenha dúvidas em como funciona a sintaxe de chamada da aplicação, o software apresenta uma ajuda geral (BAMBINETTI, 2008).

Quadro 2 - Ajuda Geral - Comandos Disponíveis

Use:	
wscmd.py [server:port] estacao	<commando> Para consultar estações
wscmd.py [server:port] sistema	<commando> Para consultar sistema operacional da estação
wscmd.py [server:port] hardware	<commando> Para consultar hardware da estação
wscmd.py [server:port] pacotes	<commando> Para consultar pacotes instalados
wscmd.py [server:port] grupos	<commando> Gerencia pacotes de software
wscmd.py [server:port] scripts	<commando> Gerencia grupos de estações
wscmd.py [server:port] monitor	<commando> Gerencia monitor de processos

Fonte: Bambinetti (2008).

Quadro 3 - Ajuda - Módulo Grupos

Use:			
cria			
criar	[grupo]	<descrição>	Cria Grupo
lista	[grupo]		Para lista as estações pertecente ao grupo
apagar	[grupo]		Apagar Grupo
add	[grupo]	<estações>	Adiciona estações ao grupo
del	[grupo]	<estações>	Remover estações de um grupo

Fonte: Babinetti (2008).

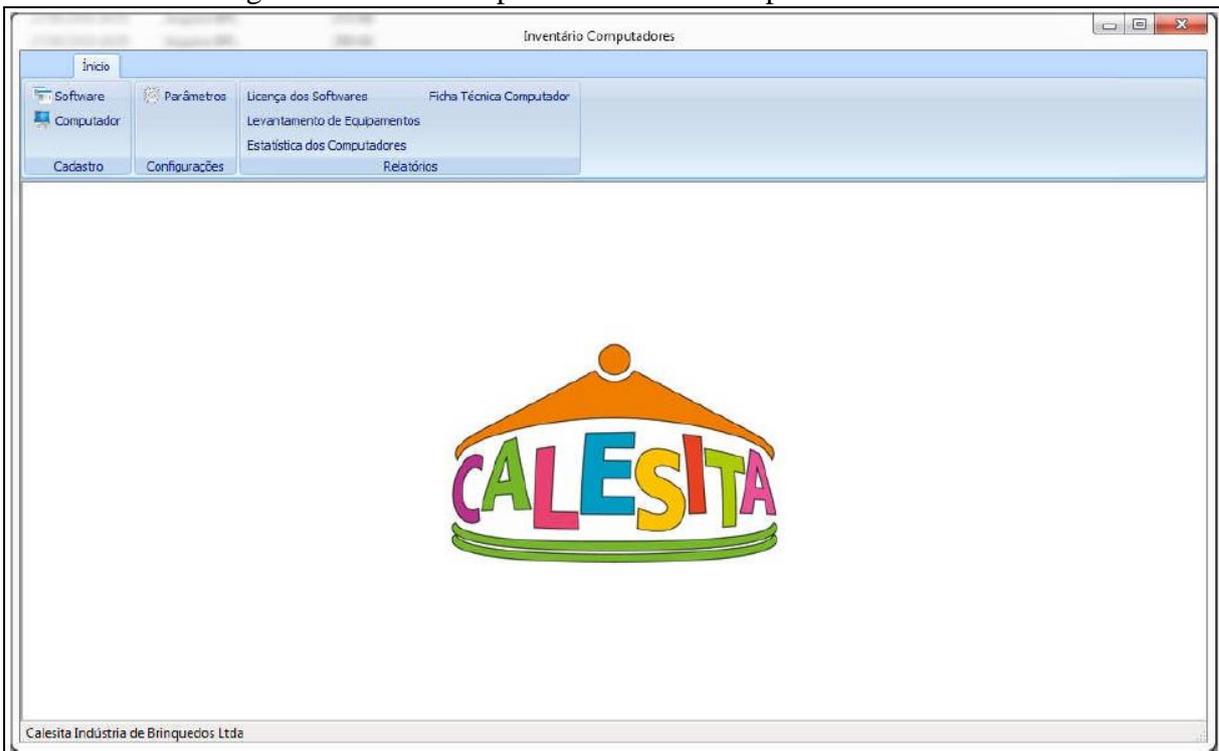
Conforme Quadro 1 se o usuário tiver necessidade de ajuda em um módulo específico é possível evocar através de comandos a ajuda de forma simplificada bastando mencionar o módulo desejado, como por exemplo o módulo grupos, demonstrado na Figura 3, que pode ser evocado através do comando `wscmd.py server:80 grupos help` (BAMBINETTI, 2008).

Em um cenário onde a aplicação seja implantada em uma rede de computadores extensa é necessário gerar automatismo pois o software não possui um instalador e desta forma exige outro formato de distribuição como por exemplo um *script*, que deve instalar os pré-requisitos e fazer *deploy* da aplicação cliente. A mesma dificuldade encontrada na instalação da aplicação é encontrada na atualização sendo necessário construção de uma funcionalidade adicional capaz de atualizar os clientes de forma automatizada e transparente aos demais usuários logados no computador auditado.

2.7.2 Sistema para inventario de computadores de redes Windows baseado na análise de registro na empresa Calesita

O software é um inventário de computadores para redes Windows, um software modular, ou seja, desconectado entre os módulos cliente e servidor, por isso sendo possível o uso mesmo fora da rede principal de computadores. O produto possui a funcionalidade de envio de e-mails que comunica o gestor da solução sob qualquer nova alteração significativa no funcionamento da solução. Um dos pontos fortes da solução é efetuar um levantamento completo da máquina, trazendo informações acerca do hardware, software, programas instalados, impressoras, enfim, qualquer característica daquela maquina inventariada. Os dados coletados são gravados previamente em Extensible Markup Language (XML) e em seguida transferidos a um diretório comum para processamento pelo módulo intermediário da solução (HEIDEN, 2006).

Figura 1 - Tela Principal - Inventário Computadores Calesita



Fonte: Heiden (2006).

O Software possui uma interface amigável conforme Figura 4, em que é possível ter todos os menus acessíveis em uma única tela e com visual limpo e fluído. O usuário se organiza e visualiza todas as opções facilmente logo no início da utilização do software que demonstra a simplicidade da utilização diante do papel importante que o software executa dentro da empresa Calesita.

Figura 2 - Listagem do Inventário de Computadores

Computador	Nome Computador	Ultima Atualização	Sistema Operacional	Service Pack	HD	Memória	Excluir
	Almoxarifado	06/09/2013	Microsoft Windows XP	Service Pack 3		74,55 GB	0,99 GB
	Almoxarifado2	05/09/2013	Microsoft Windows XP	Service Pack 3		37,30 GB	999,48 MB
	APPLICATION	09/09/2013	Windows Server 2008 R2 Standard	Service Pack 1		126,90 GB	16,00 GB
	Atendimento	20/09/2013	Windows 7 Professional	Service Pack 1		46,5,66 GB	3,88 GB
	Barbara	20/09/2013	Windows 7 Professional	Service Pack 1		46,5,66 GB	3,88 GB
	Cad-1	27/08/2013	Windows 7 Ultimate	Service Pack 1		297,99 GB	7,99 GB
	Cad-2	06/09/2013	Windows 7 Ultimate	Service Pack 1		297,99 GB	7,99 GB
	Cad-3	06/09/2013	Windows 7 Professional	Service Pack 1		46,5,66 GB	15,93 GB
	Cad-4	06/09/2013	Windows 7 Professional	Service Pack 1		931,41 GB	15,93 GB
	Cadastro	24/09/2013	Windows 7 Ultimate	Service Pack 1		297,99 GB	3,99 GB
	Cleiton	18/09/2013	Microsoft Windows XP	Service Pack 3		74,52 GB	0,99 GB
	Compras	20/09/2013	Windows 7 Professional	Service Pack 1		232,11 GB	3,88 GB
	Compras2	20/09/2013	Windows 7 Professional	Service Pack 1		297,99 GB	3,97 GB
	Contabilidade	19/09/2013	Windows 7 Professional	Service Pack 1		297,99 GB	1,99 GB
	contabilidade2	17/09/2013	Microsoft Windows XP	Service Pack 3		74,52 GB	1,99 GB
	Contabilidade3	16/09/2013	Windows 7 Ultimate	Service Pack 1		297,99 GB	7,96 GB
	DB1	09/09/2013	Windows Server 2008 R2 Enterprise	Service Pack 1		175,68 GB	63,96 GB
	Design	24/09/2013	Windows 7 Ultimate	Service Pack 1		297,99 GB	7,99 GB
	expedicao	21/09/2013	Microsoft Windows XP	Service Pack 3		74,55 GB	1,44 GB
	Faturamento	18/09/2013	Windows 7 Professional	Service Pack 1		74,43 GB	4,00 GB
	Financeiro	24/09/2013	Windows 7 Professional	Service Pack 1		232,11 GB	3,88 GB
	Financeiro2	23/09/2013	Windows 7 Professional	Service Pack 1		232,11 GB	3,88 GB
	financeiro3	17/09/2013	Microsoft Windows XP	Service Pack 3		74,52 GB	1,47 GB
	Financeiro4	20/09/2013	Windows 7 Professional	Service Pack 1		232,73 GB	1,96 GB

Fonte: Heiden (2006).

Conforme a figura 5 o software também possui um fácil acesso aos dados levantados e são necessários poucos cliques para acesso as demais telas. É possível consultar quantos computadores estão no inventário com uma apresentação já nesta mesma tela de alguns dados levantados.

O Software possui um ponto avaliado como negativo no quesito segurança, que é o software não ser protegido por uma tela de *login* com usuário e senha que impeça o uso por qualquer uma pessoa. Por não haver uma tela de login que filtraria o acesso aos dados levantados as informações levantadas ficam expostas. Por ser um produto de estudo de caso a solução é direcionada à empresa do autor, o que não permitiu uniformizar a solução, não podendo ser usada em qualquer outra empresa. Não foi definido um processo de instalação e os pré-requisitos para funcionamento devem ser feitos de forma manual. (HEIDEN, 2006).

O resultado em comparação aos objetivos estabelecidos propostos no software desenvolvido Heiden (2006) teve êxito na implementação inclusive o design da solução superou as expectativas e o supervisor afetado pela construção da ferramenta apontou satisfação e melhora como um todo no processo de controle de ativos de TI e a gerência de configuração da empresa (HEIDEN, 2006).

2.7.3 Software de inventário de software de equipamentos de rede utilizando session message block

Kreuch (2007) desenvolveu um software em continuidade a outro já existente, um software de levantamento patrimonial do parque de hardware. A motivação deste trabalho foi desenvolver um software de levantamento de inventário de softwares, complementando a software originalmente desenvolvido por OLIVEIRA JUNIOR (2005).

Como principais funcionalidades se destaca o levantamento dos softwares existentes através de um cliente que pode funcionar em estações de dentro ou de fora da rede principal do servidor da aplicação. Além disso, pode ser destacado também o envio das informações coletadas para um servidor principal que trata a informação para visualização e em seguida persiste os dados em um banco de dados Relacional MySQL. O servidor armazena localmente os dados levantados antes de persistir na base de dados. Os pontos fortes da aplicação são a coleta completa de todos os softwares instalados na estação que roda o cliente do software. O software disponibiliza relatórios dos softwares instalados. No modulo servidor existe um mecanismo de notificação ao administrador sobre alterações nas coletas do software. É possível enviar as notificações via e-mail (KREUCH, 2007).

Figura 3 - Tela principal - lado cliente



Fonte: Kreuch (2007).

Conforme figura 6 o software cliente instalado na estação auditada possui uma interface de fácil checagem das informações obtidas, centralizando todas as informações incluindo as que não são facilmente obtidas como exemplo a marca e modelo dos periféricos de rede e vídeo que costumam somente ficar acessíveis com um conhecimento avançado do ser humano para localizar as informações (KREUCH, 2007).

Figura 4 - Relatório de Softwares Inventariados

Report Preview

File Page Zoom

Page 3 of 4 Zoom 100.0 %

Relatório de Histórico de Softwares Inventariados Estações

Nome Computador: MICR01 Sistema Operacional: Windows 2000 Login: Administ

IP: 192.168.254.5 Mac Address: 00-03-FF-09-67-E2 Último acesso em: 02/05/2007 10:13:48

Nome do Software	Data/hora	Tipo- 1 Incluído 2 - Excluído
Hotfix for MDAC 2.53 (KB927779)	10/04/2007 21:57:18	1
Windows 2000 Hotfix - KB928090	10/04/2007 21:57:20	1
Windows 2000 Hotfix - KB928843	10/04/2007 21:57:21	1
Windows 2000 Hotfix - KB929969	10/04/2007 21:57:22	1
Trend Micro OfficeScan Client	10/04/2007 21:57:22	1
Hotfix do Windows Media Player (consulte Q628026 para obter mais infor	10/04/2007 21:57:23	1
SAP GUI 7.10	10/04/2007 21:57:25	1
SAP Enterprise Central Component 6.0, SR1 English	10/04/2007 21:57:25	1
Pacote Cumulativo de Atualizações 1 para o Windows 2000 SP4	10/04/2007 21:57:26	1
MSXML 4.0 SP2 (KB927978)	10/04/2007 21:57:27	1
Microsoft redistributable runtime DLLs VS2005(x86)	10/04/2007 21:57:28	1
Microsoft Internet Explorer 6 SP1	10/04/2007 21:57:29	2
SAP Front End	10/04/2007 21:57:29	2
Trend Micro OfficeScan Client	10/04/2007 22:04:25	2
Windows 2000 Hotfix - KB922582	21/04/2007 14:29:23	1
SAP Enterprise Central Component 6.0, SR1 English	21/04/2007 14:29:24	2
Microsoft redistributable runtime DLLs VS2005 SP1(x86)	21/04/2007 14:29:24	1
SAP Business Explorer	02/05/2007 09:14:38	1

Fonte: Kreuch (2007).

Além das telas de consulta o software apresenta, conforme Figura 7, possibilidade de emissão de relatórios compilando as informações geradas, seja a respeito de alguma estação específica ou de várias, concatenando as informações necessárias. As informações são tabuladas e organizadas para fácil leitura do usuário (KREUCH, 2007).

Como ponto fraco o software não trabalha de forma autônoma, ou seja, sem o servidor o cliente não efetua coletas, desta forma não é possível que seja instalado em redes distintas para levantamento de informações de parque de TI, e nem fora da rede de computadores principal do cliente. O Cliente por não funcionar desacoplado do servidor não possui um armazenamento local dos dados coletados (KREUCH, 2007).

O resultado em comparação aos objetivos estabelecidos por este trabalho, ficou demonstrado que o autor teve êxito ao complementar o software original com novo o bloco de funcionalidades voltadas a parte de software e com isso ao que foi proposta teve êxito no desenvolvimento.

3 DESENVOLVIMENTO DO SOFTWARE

Neste capítulo serão abordadas as partes relevantes do software desenvolvido. Será demonstrado detalhes da implementação e especificação através de UML, ferramenta de desenvolvimento utilizada e os resultados obtidos.

3.1 LEVANTAMENTO DE INFORMAÇÕES

Nesta etapa do trabalho foram pesquisados aspectos básicos da aplicação, por exemplo, qual plataforma e linguagem seria codificada. Por afinidade maior, se optou pela linguagem Visual Basic .NET através da IDE de desenvolvimento Visual Studio 2015. Foram feitas pesquisas acerca da versão do .NET framework ideal se descobriu que quanto mais atualizado o .NET mais necessário é usar *guests* com Windows na últimas versões de mercado devido a descontinuidade do próprio fabricante. Ficou decidido que o .NET a ser utilizado seria o 3.5 que possibilita uso de *guests* com Windows XP, Vista, 7, 8 e 10, maioria das versões Windows em funcionamento.

A partir do esboço do software, linguagem e IDE estabelecida foram feitas pesquisas nos correlatos encontrados e informações individuais sobre os obtidas através de *comand shell* que a própria Microsoft disponibiliza aos usuários e com isso obtido um escopo de funcionalidades que deveriam estar presentes de modo a garantir o maior número de informações possíveis. Nessa etapa foi analisado os trabalhos correlatos e descoberto que ainda haviam informações que não estavam presentes nos trabalhos encontrados e que poderiam ser levantadas também como por exemplo a chave de instalação do Windows e Office do computador auditado. Foi levantado que seria importante a auditoria física (memória, processador, disco, etc.) e virtual (Windows, Office, Programas Instalados, Parâmetros de usuários, etc.). Com essas informações em mãos foi possível a montagem de um escopo de possíveis entregas as quais faltava apenas avaliar se a IDE e Linguagem daria suporte tecnológico necessário.

O passo seguinte foi analisar a arquitetura mais adequada e flexível ao desenvolvimento do software, além disso se imaginou que devido ao volume de dados e a facilidade de uso esperada não poderia ficar restrito a uma arquitetura WEB. As informações seriam todas provenientes de *guests* existentes em uma rede local ou mesmo uma subrede e desta forma a WEB não seria uma arquitetura obrigatória. Diante do volume de dados possíveis de se gerar se julgou necessário ao menos ter uma forma de visualizar as informações que seriam armazenadas em banco relacional a ser definido. Surgiu a ideia de funcionamento desconectado e contínuo de tal forma que o módulo cliente não precisasse

estar conectado ao servidor para que fizesse a coleta das informações e muito menos precisasse da base relacional para persistir as informações, este último seria tarefa de um segundo agente denominado servidor. Desta forma se optou por existir um cliente instalado localmente de fácil instalação e um servidor que recebesse e organizasse as informações em um banco relacional. Para funcionamento local no *guest* se optou por não haver um banco de dados local e sim geração de um XML com as informações coletadas e que estas fossem sendo transferidas de forma simples para o servidor central e que independentemente da quantidade de XMLs existentes o servidor poderia enfileirar e tratar de forma paralela ao funcionamento da rotina de coletas, dando maior agilidade e melhor performance na solução de forma geral.

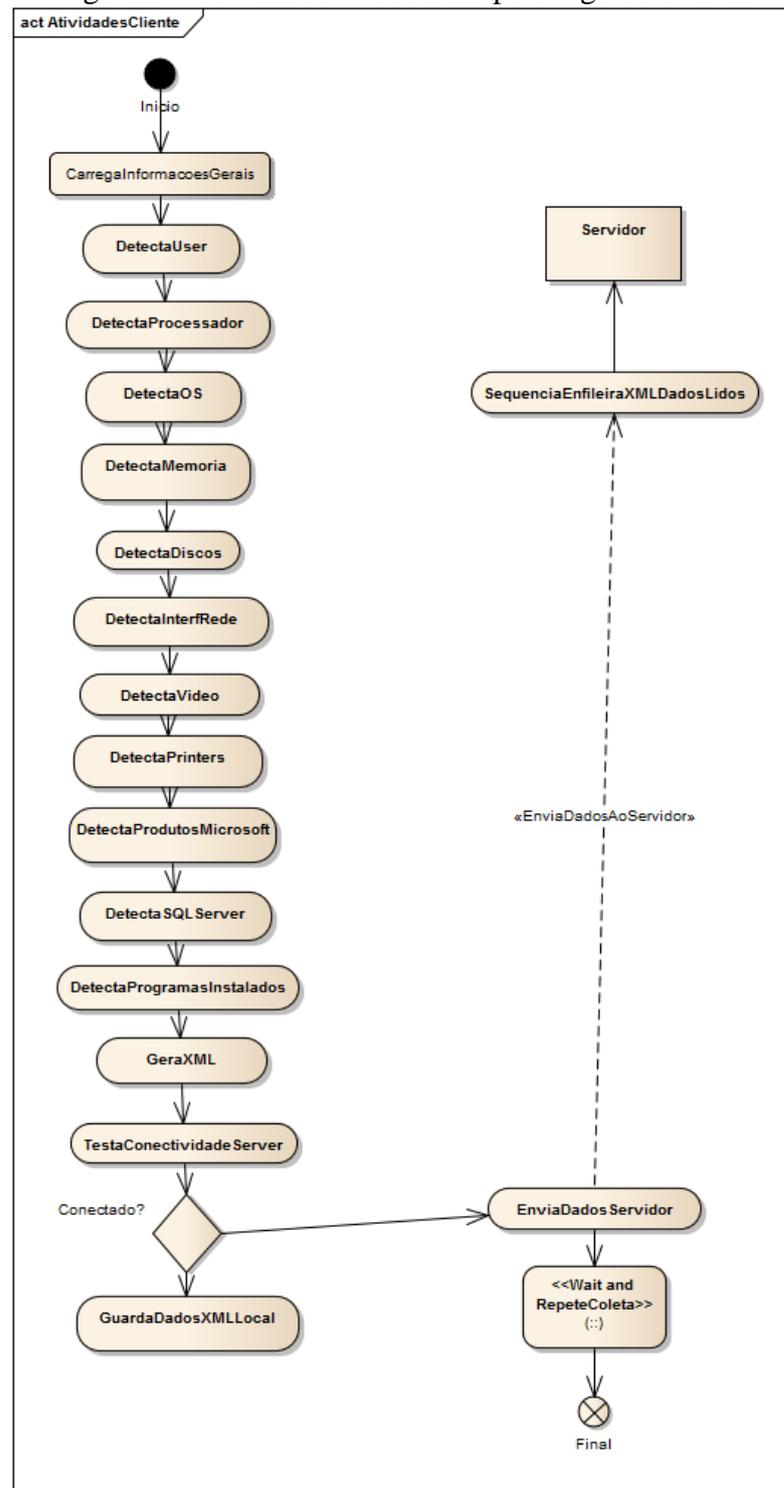
3.2 ESPECIFICAÇÃO

Para especificação foi usado UML para modelar os diagramas de Classe e Atividade. A Ferramenta usada foi o Enterprise Architect da Sparx Systems.

3.2.1 Diagrama de Atividades

A figura 8 representa o diagrama de atividades, sequenciadas a serem executadas pelo Agente-Cliente de tal forma caracterizar o funcionamento do software na parte cliente desenvolvido.

Figura 5 - Atividades Executadas pelo Agente Cliente



Fonte: Elaborado pelo Autor.

De acordo com a figura 8 existem os processos que são executados pelo cliente toda vez que inicia para que seja feito a coleta dos dados no *host*. Esta execução é recursiva para sempre estar com os dados atualizados e a periodicidade desta recursividade é um parâmetro do software. Estas atividades são:

- a) Diagnostica informações gerais para a coleta, isso inclui a data e a hora da

coleta;

- a) Detecta o Processador. Nesta fase a solução levanta informações a respeito do processador, isso inclui modelo, clock, quantidade cores existentes, fabricante);
- b) Detecta Sistema Operacional. Nesta fase a solução detecta a versão e Service Pack do Sistema Operacional;
- c) Detecta Memória RAM. O programa faz levantamento do quanto de memória RAM está instalado, o quanto está em uso e o quanto está livre;
- d) Detecta os Discos Locais. Nesta fase o programa verifica quais são os discos conectados ao *host* e destes quais são realmente locais, desconsiderando discos de Rede por exemplo;
- e) Detecta Adaptadores de Rede. O Programa levanta quantas placas de rede existem e de cada placa encontrada levanta dados como MacAddress, IPv4, IPv6;
- f) Detecta Informações de Vídeo. Nesta etapa o programa levanta informações sobre a parte de vídeo do *guest* auditado, como por exemplo resolução atual de vídeo e modelo da placa de vídeo com sua descrição;
- g) Detecta Impressoras Instaladas. Nesta etapa o programa busca todas as impressoras (locais e de rede) instaladas e as enumera;
- h) Detecta as Chaves de Instalação de Produtos Microsoft (Windows, Office e SQL). Nesta o programa busca os softwares que são do Fabricante Microsoft e Enumera a chave de instalação;
- i) Detecta Programas Instalados. O programa faz uma listagem dos programas atualmente instalados e organiza-os de forma alfabética;
- j) Gera XML. O agente faz o agrupamento das informações coletas e gera um XML no layout do software;
- k) Detecta Server. O agente executa um teste de conectividade com o servidor, uma validação se existe a comunicação. Caso exista é aberto a conexão para envio do XML gerado como também demais XMLs que não tenham sido transferidos ainda. Caso contrário é armazenado local no *guest* e na próxima execução da coleta é refeito o teste de conexão e posterior envio do XML;

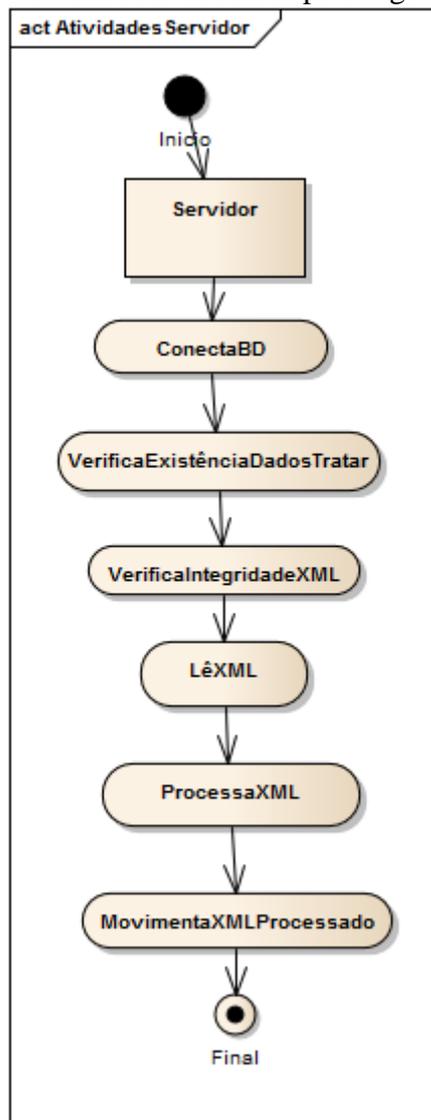
A Figura 9 representa o diagrama de atividades do agente instalado no servidor de aplicação e representa as atividades sequenciadas a serem executadas pelo software servidor

no momento do recebimento de dados direcionados pelos *hosts* auditados, enfileiramento dos arquivos recebidos e processamento contínuo de acordo com o surgimento de novas tarefas.

Existem processos que são executados somente na inicialização do serviço Windows e outros que são executados de forma recursiva aonde a cada ciclo se verifica existência de novas atividades e as executa, ficando em seguida no aguardo de novas atividades. Estas atividades de forma organizada são:

- a) Verifica se existem dados a tratar. O Servidor fica monitorando localmente uma diretório onde são armazenados os XMLs gerados pelo agente-cliente de cada máquina auditada;
- b) Executa uma rotina de validação da estrutura do arquivo XML, fazendo somente a verificação da estrutura do arquivo, se condiz com a estrutura padrão de arquivos XMLs gerados pelo software e se será possível processamento deste por parte do software. Se o arquivo estiver malformado é executado uma rotina de descarte que invalida o arquivo e o retira da fila de processamento;
- c) Caso a estrutura esteja correta é acionado a rotina `lêXML` própria de leitura do arquivo que lê linha-a-linha todas as informações do arquivo, com isso as chaves e valores lidos são trazidos para ambiente de execução, as informações levantadas ficam disponíveis em vetores para que sejam organizadas e inseridas na base de dados SQL Server;
- d) É acionada uma rotina de manipulação dos dados que faz com que estes sejam organizados em blocos de comando. Em seguida são consistidos e inseridos na base de dados SQL Server;

Figura 6 - Atividades executadas pelo Agente Servidor

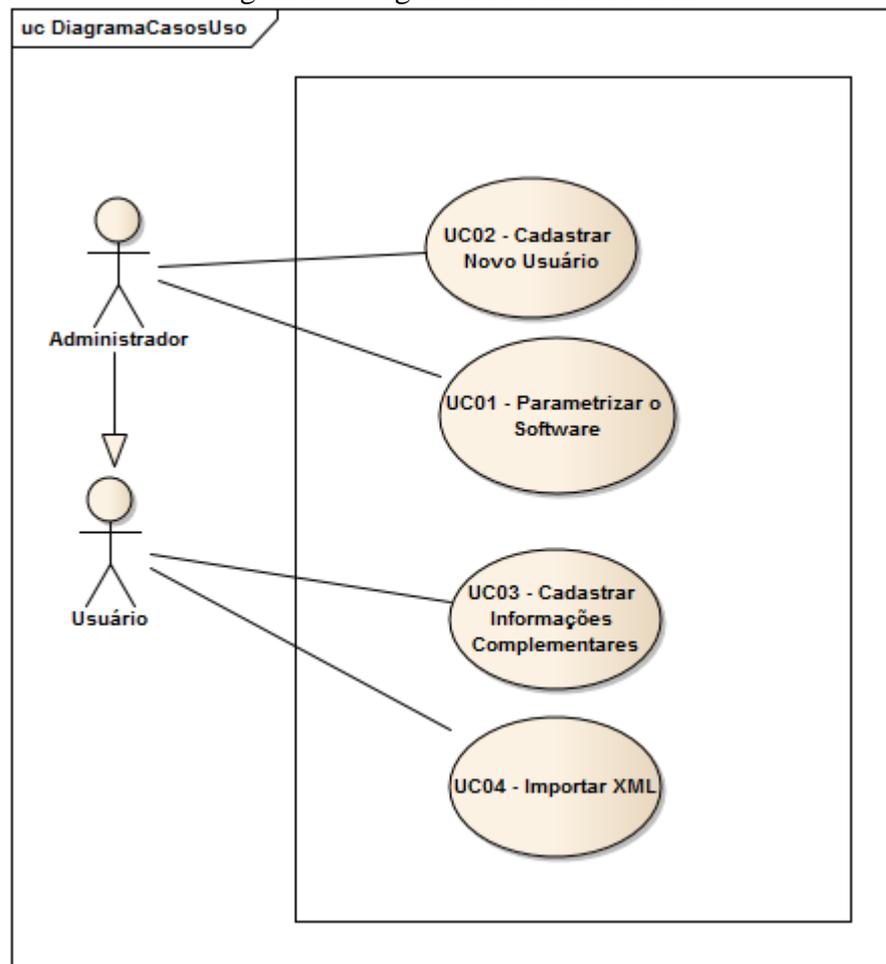


Fonte: Elaborado pelo Autor.

3.2.2 Diagrama de casos de uso

O diagrama da figura 10 representa os casos de uso existentes no software e especifica o funcionamento da parte cliente e da parte servidora para se chegar ao objetivo principal do software, o levantamento dos dados do *host* o qual está instalado e transmissão dos dados levantados. A seguir é apresentado os diagramas de caso de uso do usuário administrador e do usuário comum, atores que interagem com o software durante a operação.

Figura 7 - Diagrama de Casos de Uso



Fonte: Elaborado pelo Autor.

Existem dois atores principais, que interagem com o software. O primeiro ator é administrador da solução que fará após instalação as parametrizações do arquivo inicializador (auditco.ini) e além disto fará o cadastro de novos usuários. O segundo ator é o usuário que acessará a aplicação e fará a dita operação através de importação do XML avulso e também cadastro de informações complementares dos *guests* auditados.

A seguir são listados os casos de uso utilizados pelo software de forma detalhada conforme figura 10:

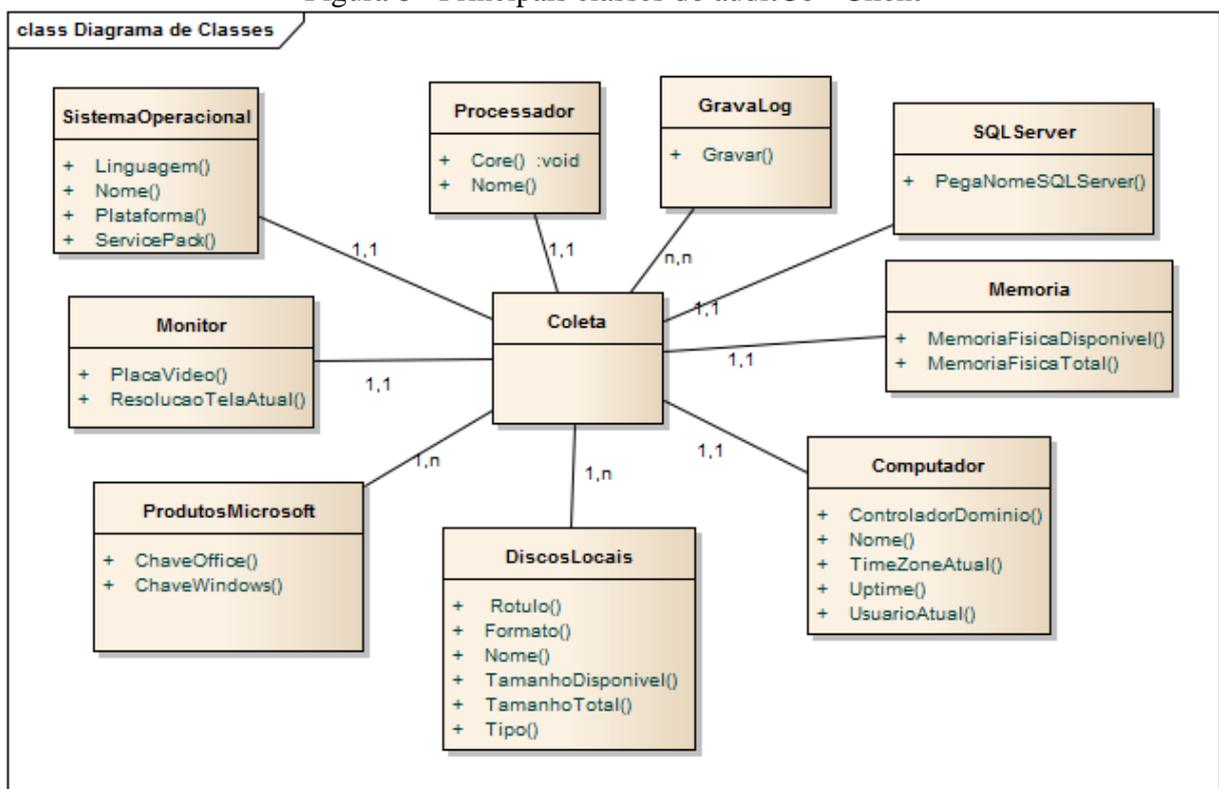
- a) UC01 - parametrizar o software: o administrador atribui valores personalizados para os parâmetros-chaves de funcionamento do software como o diretório onde serão transferidos os XML com os dados coletados de cada host, endereço de encaminhamento dos XMLs, usuário e senha padrão para acesso ao diretório da solução.
- b) UC02 - cadastra novo usuário: o administrador cadastra novo usuário conforme os campos demonstrados na tela de cadastro e lhe atribui uma senha para uso do software.

- c) UC03 - cadastra informações complementares: o usuário atribui a um host auditado informações complementares como NF de aquisição, data de aquisição e setor predial de localização do *host*.
- d) UC04 - importa XML: o usuário em poder de um ou mais XML gerados pelo software faz o upload manual dentro do software integrando informações pendentes coletadas de forma manual.

3.2.3 Diagrama de Classes

A seguir é apresentado o diagrama de classes que estrutura o software desenvolvido demonstra as principais classes utilizadas dentro do software. De acordo com a Figura 11, a classe *Coleta* é a principal do software desenvolvido sendo utilizada unicamente no software cliente, estruturando o levantamento, tratamento e persistência dos dados do início ao fim da execução das rotinas. Esta classe possui chamadas para as demais funções desenvolvidas.

Figura 8 - Principais classes do auditCo - Client



Fonte: Elaborado pelo Autor.

3.3 IMPLEMENTAÇÃO

A seguir é descrito o detalhamento da implementação do software desenvolvido conforme as especificações explicadas neste documento.

3.3.1 Técnicas e ferramentas utilizadas

Para codificação do software cliente e servidor foi utilizado a IDE Visual Studio 2015 utilizando a linguagem de programação visual basic .NET, ambas propriedades da Microsoft ®. O visual basic utiliza o *framework* .NET que possui recursos que facilitam o acesso e a manipulação de informações que o próprio Windows demonstra de forma individual quando solicitado pelo usuário através de *shell command* ou visualizado de forma individual e/ou agrupada no explorador de informações do computador o qual tem o Windows instalado.

No quadro 1 é mostrado trecho da função `pegaParametro` membro do componente `funcoes.vb` que contém toda a implementação da biblioteca de classes e funções desenvolvidas para o software. A função faz a leitura do arquivo `.ini` que contém os parâmetros personalizados pelo administrador do software, com esta função basta existir um arquivo com o nome `AuditcoClient.ini` conforme exemplo que a função fará a abertura e leitura deste atrás da sessão e parâmetro solicitados pelo usuário.

Quadro 4 – Procedimento que lê parâmetros através de arquivo INI

```
Public Function PegaParametro(ByRef CDESSAO As String, ByRef CDPARAM As String) As String
    Dim NRTAMPAR As Integer = 0
    Dim DSPARAME As String = New String(Chr(0), 255)
    Dim NMARQINI As String = My.Application.Info.DirectoryPath & "\AuditcoClient.ini"
    NRTAMPAR = GetPrivateProfileString(CDESSAO, CDPARAM, "", DSPARAME, Len(DSPARAME), NMARQINI)
    PegaParametro = Microsoft.VisualBasic.Left(DSPARAME, NRTAMPAR)
End Function
```

Fonte: Elaborado pelo Autor.

O mesmo arquivo `funcoes.vb` é utilizado na parte servidora como a biblioteca de classes e funções utilizadas pelo software para executar a manipulação e persistência das informações na base de dados. Da mesma forma que o cliente faz a coleta dos dados e formata o arquivo XML, o servidor faz também a leitura no mesmo formato e caso haja alguma inconsistência descarta aquela leitura recebida.

No quadro 2 tem-se o trecho do código de ativação da coleta dos dados, existente no serviço *cliente* do software. O trecho `OnStart` é a parte do código de um serviço Windows que é executado após inicialização do serviço. No trecho é visto que o software pega um parâmetro personalizado chamado `TempoColeta`, o qual possui um valor determinando o tempo de espera entre uma coleta e outra, em seguida executa uma verificação se é necessário gerar fisicamente o XML ou se apenas deve ser coleta os dados e enviados de forma *online* ao servidor. Posteriormente ele executa a coleta e entra em estado de descanso aguardando a próxima vez que deve executar.

Quadro 5 - Trecho do Código OnStart - Serviço Cliente

```
Protected Overrides Sub OnStart(ByVal args() As String)
    Try
        Do While True
            Thread.Sleep(tempoColeta)
            ExecutaColeta()
            If PegaParametro("GERAL", "CRIAXML") = "SIM" Then
                CriaXML()
            End If
        Loop

        Catch ex As Exception
            GravaLog("Erro de Execução Serviço: " & ex.Message)
            GravaLog("Erro de Execução Serviço: " & ex.Source)
            GravaLog("Erro de Execução Serviço: " & ex.HelpLink)
            GravaLog("Erro de Execução Serviço: " & ex.StackTrace)
        End Try
    End Sub
```

Fonte: Elaborado pelo Autor.

A estrutura de inicialização do serviço cliente é feita de forma simplificada, conforme demonstrado no Quadro 3, onde somente é necessário a leitura do parâmetro `tempoColeta` e `criaXML` os quais definem o tempo em milissegundos que será executado a rotina de coleta e se é desejado que o software crie um XML da coleta executada.

O software possui como principal função a criação do XML a partir da coleta feita. No quadro 4 tem-se parte do código da função que cria o XML após a coleta e podemos observar que o software segue uma formatação padrão e instancia as demais funções para que receba a informação necessária e formate da forma padrão no formato XML.

Quadro 6 - Parte do código da função CriaXML

```

Public Function CriaXML()
    Dim ContaImpressoras As Integer = 0
    Dim ContaProgramas As Integer = 0
    MontaComando(DataHoje, ",")
    MontaComando(HoraAtual, ",")
    Dim xmlPath As String = "C:\AuditCoClient\XML\" & Date.Now.Year & Date.Now.Month
    ' Cria um novo ficheiro XML com a codificação UTF8
    Dim xmlw As New XmlTextWriter(xmlPath, System.Text.Encoding.UTF8)
    xmlw.Formatting = Formatting.Indented

    xmlw.WriteStartDocument()

    ' Adiciona um comentário geral
    'xmlw.WriteComment("Declaracao do Node Pai")

    ' Criar um elemento geral
    xmlw.WriteStartElement("AuditCo_Cliente")

    ' Criar o Node Filho "Computador" e alguns dados
    With xmlw
        ' Adiciona um comentário Identificador Node Filho
        'xmlw.WriteComment("Declaracao do Node Filho Computador")
        .WriteStartElement("Computador")
        .WriteElementString("Nome", Computador.Nome)
        MontaComando(Computador.Nome, ",")
        .WriteElementString("Usuario", Computador.UsuarioAtual)
        MontaComando(Computador.UsuarioAtual, ",")
        .WriteElementString("ControladorDominio", Computador.ControladorDominio)
        MontaComando(Computador.ControladorDominio, ",")
        .WriteElementString("Uptime", Computador.Uptime)
        MontaComando(Computador.Uptime, ",")
        .WriteElementString("TimeZone", Computador.TimeZoneAtual)
        MontaComando(Computador.TimeZoneAtual, ",")
    End With
End Function

```

Fonte: Elaborado pelo Autor.

Conforme mostra o quadro 3, depois que a coleta é executada é verificado se o parametro de `criaXML` está ativo e se estiver um XML desta coleta todo conteúdo obtido pelo cliente é enviado na integra ao servidor caso haja conexão disponível. Caso não este arquivo fica armazenado localmente esperando a próxima coleta e envio junto aos demais arquivos XML que por ventura existirem também. A seguir no quadro 5 tem-se um trecho do arquivo resultante do processo de criação do XML.

Quadro 7 - Parte do XML gerado após uma coleta

```

<discosohost>C:\</discosohost>
<formatdiscosohost>NTFS</formatdiscosohost>
<tamtotdiscosohost>298GB</tamtotdiscosohost>
<tamdispdiscosohost>196GB</tamdispdiscosohost>
<discosohost>E:\</discosohost>
<formatdiscosohost>NTFS</formatdiscosohost>
<tamtotdiscosohost>30GB</tamtotdiscosohost>
<tamdispdiscosohost>22GB</tamdispdiscosohost>
<nomeniccabo>Conexão local</nomeniccabo>
<descniccabo>Broadcom NetXtreme 57xx Gigabit Controller</descniccabo>
<macaddressniccabo>A41F72F7ECC6</macaddressniccabo>
<enderipniccabo>169.254.148.84</enderipniccabo>
<nomenicsemfio>Conexão de Rede sem Fio</nomenicsemfio>
<descnicsemfio>Adaptador de rede Broadcom 802.11n</descnicsemfio>
<macaddressnicsemfio>342387088762</macaddressnicsemfio>
<enderipnicsemfio>192.168.0.13</enderipnicsemfio>
<placavideo>Intel(R) HD Graphics 4000 com 2GB Dedicado de RAM</placavideo>
<resolucaoatual>1366 X 768</resolucaoatual>
<impressorainst1>Send To OneNote 2016</impressorainst1>
<impressorainst2>PDFCreator</impressorainst2>
<impressorainst3>PDF Architect 4</impressorainst3>
<impressorainst4>Microsoft XPS Document Writer</impressorainst4>
<impressorainst5>Microsoft Print to PDF</impressorainst5>
<impressorainst6>Fax</impressorainst6>
<impressorainst7>EPSON XP-211 214 216 Series</impressorainst7>
<impressorainst8>Epson Wifi - Casa</impressorainst8>
<chavewindows>TY4CG-JDJH7-VJ2WF-DY4X9-HCFC6</chavewindows>
<chaveoffice>N/A</chaveoffice>

```

Fonte: Elaborado pelo Autor.

O arquivo XML possui a referência do campo a qual se refere a informação e o seu respectivo valor, que foi coletado de forma autônoma pelo cliente do software. De acordo com a quantidade de dados obtidos o XML é montado com mais ou menos campos e isso faz com que o tamanho do arquivo gerado seja de tamanho variado. O Software é flexível o suficiente para se adaptar às informações obtidas, ou seja, na eventualidade de alguma informação não for coleta o XML será composto por uma única linha acompanhada do valor N/A como foi o caso do exemplo acima que o software não teve dados para atribuir à ao campo chaveoffice.

Quadro 8 - Função Gravalog

```

Public Sub GravaLog(ByVal Mensagem As String)
    If System.IO.File.Exists(NMARQLOG) Then
        ESCRITOR = New System.IO.FileStream(NMARQLOG, IO.FileMode.Append, IO.FileAccess.Write)
    Else
        ESCRITOR = New System.IO.FileStream(NMARQLOG, IO.FileMode.CreateNew, IO.FileAccess.Write)
    End If
    ESCRITORTEXO = New System.IO.StreamWriter(ESCRITOR)
    ESCRITORTEXO.WriteLine(Mensagem)
    ESCRITORTEXO.Close()
End Sub

```

Fonte: Elaborado pelo Autor.

No quadro 6 tem-se o código da função `GravaLog` a qual faz a gravação de qualquer mensagem solicitada em tempo de execução, é esta função que grava no arquivo de log os erros e mensagens programadas para acontecer em tempo de execução do software. Esta função faz com que o software seja dotado de avisos que podem ser monitorados pelo departamento de TI com qualquer ferramenta de monitoramento de arquivo texto bastando a filtragem neste arquivo. Esta função é a mesma executada na parte servidora do software e dotada dos mesmos mecanismos de monitoramento, facilitando a análise de qualquer erro que venha a ocorrer na operação do software.

Quadro 9 - Função de envio de e-mail

```
Public Sub EnviaEmail(ByVal ENDREMEX As String,
                    ByVal ENDDSTX As String,
                    ByVal DSUSUARI As String,
                    ByVal DSSENHAX As String,
                    ByVal ENDSMTPX As String,
                    ByVal DSPORTAX As Integer)
    Dim Mensagem As Net.Mail.MailMessage = New MailMessage(ENDREMEX, ENDDSTX)
    Dim AuthenticationCredential As Net.NetworkCredential = New Net.NetworkCredential(DSUSUARI, DSSENHAX)
    Dim ClienteSMTP As Net.Mail.SmtpClient = New Net.Mail.SmtpClient(ENDSMTPX, DSPORTAX)
    Mensagem.IsBodyHtml = True
    Mensagem.Attachments.Add(New Attachment(My.Application.Info.DirectoryPath & "\AuditCO" & Date.Now.Year & Date.Now.Month & Date.Now.Day
    Mensagem.Subject = "Programa AuditCO - Status Report"
    Mensagem.Body = "<P>Bom dia</P>" &
                    "<P><B><FONT COLOR=""#FF0000"">Esta é uma Mensagem Automática. Favor Não Responder.</FONT></B></P>" &
                    "<P>Segue Anexo o Log de Execução do Programa AuditCO, no Servidor: <B>" & My.Computer.Name & "</B> .Favor analisar o"
                    " e, na constatação de qualquer anomalia comunicar o Suporte.</P>" &
                    "<BR>" &
                    "<P>Obrigado.</P>"
    ClienteSMTP.EnableSsl = True
    ClienteSMTP.UseDefaultCredentials = False
    ClienteSMTP.Credentials = AuthenticationCredential
    Try
        ClienteSMTP.Send(Mensagem)
    Catch ex As Exception
        GravaLog("Erro : " & Err.Number & " - " & Err.Description)
    End Try
End Sub
```

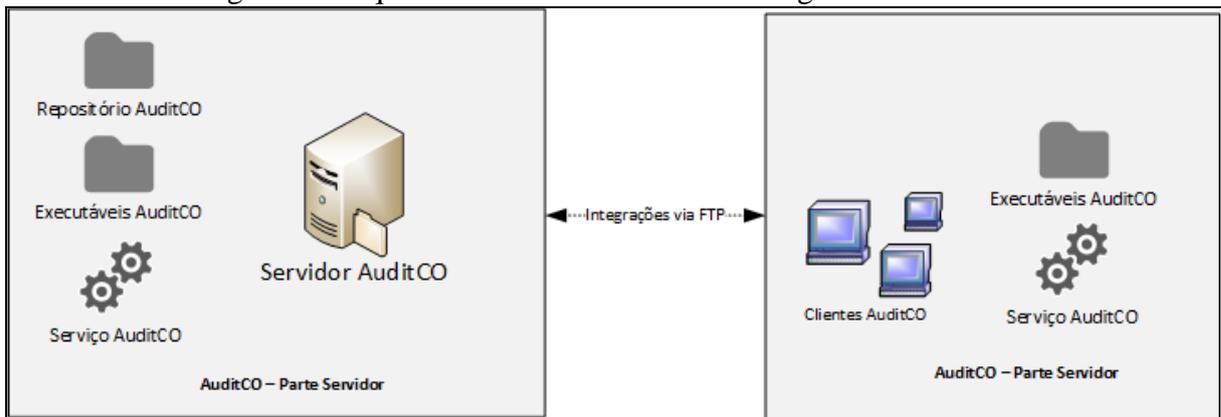
Fonte: Elaborado pelo Autor.

O software possui uma função utilizada para *status report* notificando via email quando solicitado o administrador do software com o detalhamento de como a solução está operando. No quadro 6 temos o código que faz o envio deste e-mail com um texto padronizado e com a possibilidade de anexar o log de uso da solução no envio deste e-mail.

3.3.2 Operacionalidade da implementação

O software auditCO é dividido em duas partes: cliente e servidor. O Cliente são os *notebooks*, *desktops* e servidores os quais o administrador do software deseja auditoria. O servidor é onde estarão o repositório dos arquivos recebidos e de onde serão transmitidos os arquivos de troca, ou seja, onde estarão por exemplo os arquivos de troca de versão.

Figura 9 - Esquema de funcionamento entre o guest e o servidor



Fonte: Elaborado pelo Autor.

A estrutura do software, conforme mostrado na Figura 12 tanto na parte cliente bem como na parte servidora possuem um diretório principal que abriga os executáveis e o repositório local dos XML gerados.

Figura 10 - Serviço AuditCO - Cliente instalado e pronto para o uso.

Áudio do Windows	Gerenc...	Em Exe...	Automático	Serviço local
AuditCo-Cliente			Automático	Sistema local
Autenticação Natural	Serviço...		Manual (Início do Ga...	Sistema local

Fonte: Elaborado pelo Autor.

Conforme Figura 13 a parte cliente que roda nas máquinas ditas estações e servidores também possuem um serviço Windows com inicialização logo na inicialização do sistema operacional. O software automaticamente após iniciado irá ler os parâmetros de uso contidos no arquivo `auditco.ini` e fará o início da execução do software fazendo a primeira coleta, gerando o arquivo.

Em relação à parte servidora, o software também possuirá um serviço Windows pelo qual funcionará em segundo plano, ele ficará monitorando a área de envio dos XMLs gerados pelos clientes e fará o tratamento e movimentação destes arquivos para novos diretórios de arquivamento.

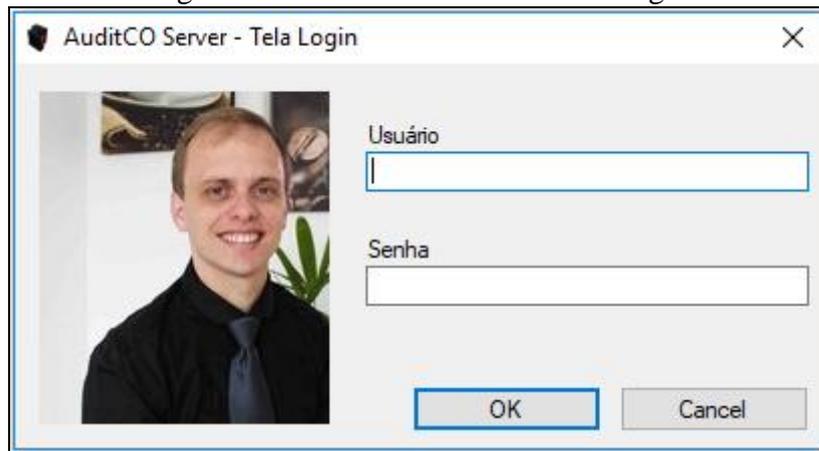
Figura 11 - Serviço AuditCO - Server instalado e pronto para o uso.

Áudio do Windows	Gerencia á...	Em Exe...	Automático	Serviço local
auditCo-Server			Automático (Atraso ...	Sistema local
Autenticação Natural	Serviço de...		Manual (Início do Ga...	Sistema local

Fonte: Elaborado pelo Autor.

Ao abrir o software na parte servidora é apresentada a tela de login solicitando o usuário e senha e em seguida é aberto a tela principal do software a qual agrupa todas as funções.

Figura 12 - AuditCo Server - tela de login



Fonte: Elaborado pelo Autor.

Ao ser aberta, conforme Figura 15 a tela de login mostra a foto do último usuário logado e os campos para digitação do usuário e senha. Desta forma é possível saber visualmente quem logou pela última vez. A tela de login impede o acesso indevido ao software na parte servidora, não deixando ocorrer vazamento dos dados.

Figura 13 - AuditCo server - tela principal



Fonte: Elaborado pelo Autor.

A Figura 16 mostra a tela principal do software pelo lado servidor. A seguir serão mostradas as guias desta tela principal. Na guia `usuários` existe a tela de cadastro de

usuários, conforme demonstra a Figura 17, desta forma é onde o operador do software informa um nome para o novo usuário e atribui um papel a ele, em seguida é necessário a escolha de uma foto deste usuário podendo ser localizada de forma simplificada através de o botão `localizar foto`. Estando o cadastro completo pode-se submeter os dados e deste momento em diante já existirá o acesso ao usuário cadastrado.

Figura 14 - Tela de cadastro de usuários

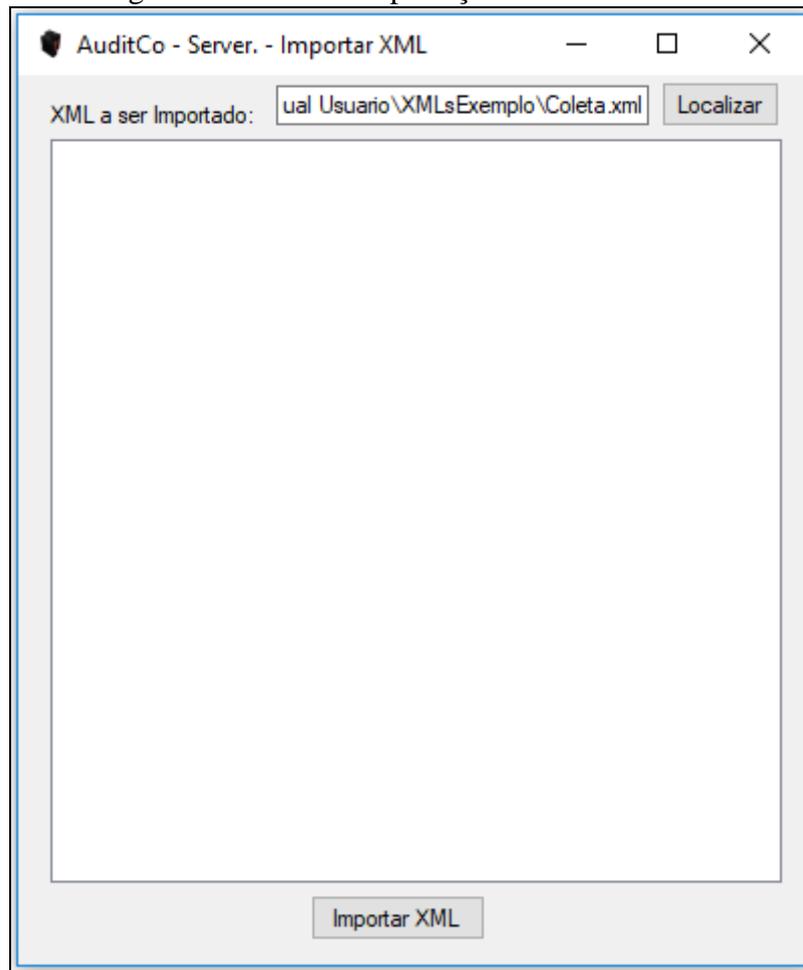


A captura de tela mostra uma janela de software com o título "AuditCO - Server. - Cadastro de Usuár...". O formulário contém os seguintes elementos:

- Nome Usuário:
- Tipo Usuário:
- Foto usuário: 
- Botão Localizar Foto
- Botão Cadastrar

Fonte: Elaborado pelo Autor.

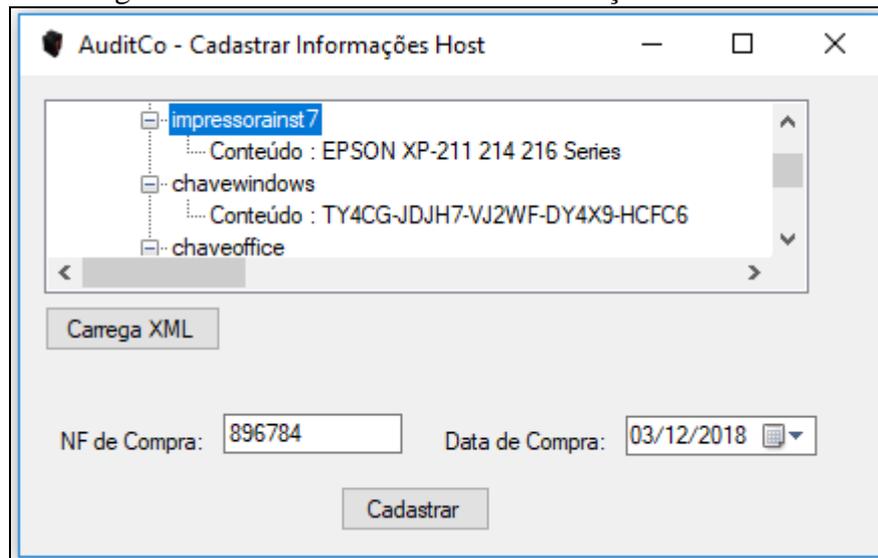
Figura 15 - Tela de importação manual de XML



Fonte: Elaborado pelo Autor.

De acordo com a Figura 18 na guia computadores fica disponível a opção de importação manual do XML gerado pelo cliente, para aquelas localidades que não possuem interligação via rede LAN ou WAN com o servidor do software. É através desta tela que a equipe poderá importar os dados de forma simples e unificada para dentro da base de dados relacional. Para uso desta tela a única informação necessária é localização do arquivo que pode estar disponível localmente no computador do usuário do software ou via rede. Para importação é necessária a localização deste arquivo e em seguida submeter a importação.

Figura 16 - Tela de Cadastro de Informações Adicionais



Fonte: Elaborado pelo Autor.

Conforme figura 19 ainda na guia `computadores` tem-se a opção de cadastro das informações de aquisição dos hosts. Para isso o usuário deve durante o *setup* do *guest* inicializar o auditCO efetuando uma coleta manual dos dados e depois, em posse do XML gerado importá-lo adicionando a informação da NF e data de compra daquele *guest*. Para cadastrar finalmente basta submeter dentro desta tela *que a partir deste momento todas as coletas são automaticamente atribuídas aquele guest*.

3.4 RESULTADOS E DISCUSSÕES

O trabalho proposto originou um software de levantamento de informações para prevenção e auditoria de software. Em complemento o software serve como agente monitorador de alterações de hardware, agrupando os dados levantados em uma base relacional de forma a deixar o levantamento organizado e de fácil acesso em caso de auditoria. Na tabela 1 é feita uma relação de funcionalidades e características entre os softwares correlatos.

Quadro 10 - Comparativo de funcionalidades com os softwares correlatos

Característica/Funcionalidade	Pamplona (2018)	Kreuch (2006)	Bambinetti (2008)	Heiden (2013)
Alertas	Não	Não	Não	Sim
Controle Licença	Sim	Não	Não	Sim
Levantamento Computadores	Sim	Sim	Não	Sim
Relatório	Sim	Sim	Não	Sim
Gera XML Levantamento	Sim	Não	Não	Sim
Banco de Dados Relacional	SQL Server	MySQL	PostgreSQL	SQL Server
Linguagem Programação	Vb.NET	Delphi	Python	Delphi
Protocolo Transmissão	FTP/POP	SMB	Webservice	TCP/IP e POP
Módulo de Troca de Versão	Sim	Não	Não	Não

Fonte: Elaborado pelo Autor.

Existe similaridade de algumas funções e características do software desenvolvido com os trabalhos correlatos, isso é evidente quando após análises detectaram-se que as necessidades de auditoria são similares, porém as formas de se fazer são diversas tendo em vista que existem diferenças claras nas possibilidades obtidas através da escolha da linguagem, plataforma e forma de desenvolvimento. O software vem para suprir uma automatização necessária, um meio comum de obter informações sobre a rede de computadores Windows e os seus softwares instalados e com isso se espera que o software seja uma fonte de consulta para a decisão para um gestor de TI da empresa em relação a auditorias externas esteja enfrentando.

Tabela 1 - Cenários de teste

Cenário de Teste	Resultado Obtido
Execução do cliente sem enviar os dados ao servidor.	100%
Obtenção de chave do Windows e Office(versão diferente da 365)	100%
Obtenção da Chave do Office 365	0%
Obtenção do nome das instâncias SQL Server instaladas em um <i>guest</i> com Windows server 2012 e SQL Server Express 2012	100%
Listagem de todas as Impressoras Instaladas, de rede ou local	100%
Listagem dos discos rígidos com capacidade total, capacidade restante e unidade de rede atribuída pelo Windows	100%
Geração do XML formatado conforme modelo a ser lido pelo servidor	100%

Fonte: Elaborado pelo Autor.

Conforme Tabela 1 dentre a bateria de testes executadas em laboratório foram detectados alguns pontos tidos como limitações, como por exemplo, o produto precisar de um servidor FTP para hospedagem dos dados e a comunicação somente possível desta forma, poderia haver uma segunda forma de integração como, por exemplo, *webservice* o qual funcionaria como *backup* da forma principal. Outra limitação observada é ser somente disponível para a plataforma Windows e caso exista algum nicho de auditoria em plataforma Linux o cliente para auditoria não está apto a isso.

A principal vantagem do uso do software se dá na implantação em redes LAN de pequena e média quantidade onde facilmente através de scripts é possível fazer o *deploy* da aplicação e a manter de forma contínua inclusive se tratando de troca de versões que são disponibilizadas em diretório do FTP que podem ser alterados através de parametrização e que os *guests* devem possuir acesso de leitura e escrita.

4 CONCLUSÕES

O Auditco conforme testes em laboratório alcançou o objetivo principal de levantar as informações sobre todo os softwares instalados no *guest*, anteriormente no quadro 4 o XML gerado possui toda a configuração do e ainda alcançou os objetivos específicos de conseguir detectar os softwares que possuem chave de instalação e os manter controlados, como exemplo o Windows, gerando insumos de decisão ao gestor. O software atendeu a eficiência esperada nos testes de coleta dos dados, montagem e envio das informações ao servidor. Em um cenário de testes utilizando o novo Office 365 o software apenas detectou a existência do software, mas não capturou sua chave. Nesse sentido a taxa de assertividade no levantamento dos dados na parte cliente ficou entre 97% a 99% dos dados, não alcançando os 100% esperados pois o licenciamento online por parte da Microsoft retirou a informação local da chave de instalação, que passou a ser validada de forma online através de credenciais de email do usuário do *guest* auditado e não precisando mais ser inserida manualmente.

O servidor por sua vez, desempenhou de forma ótima sua função, conseguiu em todos os cenários ler os arquivos enviados e persisti-los na base de dados, mesmo alguma informação não sendo possível captura foi possível gravar uma informação de dados não recebido não impedindo a persistência das demais informações no banco de dados.

Em relação às ferramentas utilizadas ficou estabelecido pós-testes que a integração de envio das informações usando o protocolo File transport Protocol (FTP) como protocolo de transporte dos dados poderia ser otimizado usando webservices que é uma das formas de integração mais usadas com softwares ERP e com isso acelerado um pouco mais o desenvolvimento e gerado novas funcionalidades para a biblioteca de código.

Como um resultado geral o software proporcionou o surgimento uma biblioteca de funções que pode ser adicionada e instanciada por qualquer outro software desenvolvido, facilitando a melhoria do atual software e novas funcionalidades.

4.1 EXTENSÕES

Como extensão do software desenvolvido seria interessante ter um aplicativo para dispositivos móveis o qual o administrador pudesse consultar as informações obtidas das auditorias feitas. Outra extensão é criar uma API no Microsoft Azure onde o aplicativo móvel irá se comunicar e consumir os dados podendo ser consultados e listados informações de todo o processo de auditoria executado. Além disso, uma extensão de funcionalidade para o programa servidor é interação com um sistema de chamados de suporte. Os chamados seria abertos automaticamente após rodar uma série de agendamentos de processos que fariam o

cruzamento das licenças cadastradas com as licenças levantadas e caso encontrado alguma inconsistência, por exemplo duas ou mais máquinas com mesma licença Windows então seria lançado uma pendência e assim seria gerado um subprocesso de abertura de chamado com detalhes do problema encontrado. Por fim deveria ser criado um cliente para que rode em Linux para auditar demais máquinas da rede e assim ter uma abrangência maior na auditoria executada.

REFERÊNCIAS

- BAMBINETI, Charles. **Sistema de Web Service para Inventário de Estações em Rede**. 2008. 49f. Trabalho de Conclusão de Curso (Bacharel em Ciência da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- TECLÓGICA. **O uso da tecnologia no Controle de Ativos de TI das empresas. Blumenau-SC. 2012**. Disponível em < <https://blog.teclogica.com.br/o-uso-da-tecnologia-no-controle-de-ativos-de-ti-das-empresas/>>. Acesso em 10 ago. 2018
- COSTA, Rodrigo Gonçalves Porto da; RIBEIRO, Sérgio Ferreira. **Desvendando e dominando o registro do Windows**. Rio de Janeiro : Ciência Moderna, 2004. xii, 158 p, il.
- FREITAS, Marcos André dos Santos. **Fundamentos do gerenciamento de serviços de TI: preparatório para a certificação ITIL V3 Foundation**. Rio de Janeiro : Brasport, c2010. xvi, 351 p, il., grafs.
- HEIDEN, Rodrigo José. **Sistema para inventário de computadores de redes Windows baseado na análise de registro na empresa Calesita**. 2013. 55 f., il. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2013. Disponível em: <http://www.bc.furb.br/docs/MO/2013/363952_1_1.pdf>. Acesso em: 12 ago. 2018.
- HF TECNOLOGIA. **Fiscalização contra pirataria aumenta nas pequenas empresas (Parte 1). Curitiba-PR. 2018**. Disponível em < <https://hftecnologia.com.br/fiscalizacao-contra-pirataria-aumenta-nas-pequenas-empresas-parte-1>>. Acesso em 23 ago. 2018
- HF TECNOLOGIA. **Fiscalização contra pirataria aumenta nas pequenas empresas (Parte 2). Curitiba-PR. 2018**. Disponível em < <https://hftecnologia.com.br/fiscalizacao-contra-pirataria-aumenta-nas-pequenas-empresas-parte-2>>. Acesso em 23 ago. 2018
- KREUCH, Josélio. **Software de inventário de software de equipamentos de rede utilizando session message block**. 2007. 45 f, il. Trabalho de Conclusão de Curso - (Graduação em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2007. Disponível em: <http://www.bc.furb.br/docs/MO/2008/329181_1_1.pdf>. Acesso em: 12 ago. 2018.
- MACORATTI, José Carlos. **VB .NET - Obtendo a chave para um produto Microsoft Instalado. São Paulo-SP. 2010**. Disponível em < http://www.macoratti.net/11/10/vbn_chv1.htm >. Acesso em 14 set. 2018.
- MAGALHÃES, Ivan Luizio; PINHEIRO, Walfrido Brito. **Gerenciamento de serviços de TI na prática: uma abordagem com base na ITIL, inclui ISO/IEC 20.000 e IT Flex**. São Paulo : Novatec, 2007. 667 p, il. (Gerenciamento de TI).
- MICROSOFT DOCUMENTS. **Introdução ao .NET Framework. São Paulo-SP. 2018**. Disponível em <<https://docs.microsoft.com/pt-br/dotnet/framework/get-started/>>. Acesso em 13 set. 2018.
- OLIVEIRA JUNIOR, Edmundo Nolas de. **Protótipo de software para gerência de patrimônio dos equipamentos de uma rede utilizando session message block**. 2005. 52 f, il. Trabalho de Conclusão de Curso - (Graduação em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2005. Disponível em: <http://www.bc.furb.br/docs/MO/2005/292152_1_1.pdf>. Acesso em: 28 nov. 2018.

ORRICO JÚNIOR, Hugo. **Pirataria de Software**. São Paulo : MM Livros, 2004. 230p.

PORTAL ABES. **Saiba mais sobre pirataria de software**. São Paulo-SP. 2018. Disponível em < <http://www.abessoftware.com.br/propriedade-intelectual/saiba-mais-sobre-pirataria-de-software>>. Acesso em 09 ago. 2018.

PORTAL GSTI. **O que é COBIT**. São Paulo-SP. 2018. Disponível em <<https://www.portalgsti.com.br/sobre>>. Acesso em 13 de out. 2018.

ROMAN, Steven; PETRUSHA, Ronald; LOMAX, Paul. **Linguagem VB NET: o guia essencial**. Rio de Janeiro : Campus, 2002. xiv, 699p.

SARTORE, Décio. **Auditoria de Software - Cada dia mais frequente nas empresas. Qual a base legal e como proceder**. São Paulo-SP. 2014. Disponível em < <https://dasartore.jusbrasil.com.br/artigos/161941641/auditoria-de-software-cada-dia-mais-frequente-nas-empresas-qual-a-base-legal-como-proceder>>. Acesso em 24 ago. 2018.

TARIFA, Alexandre; FACUNTE, Emerson; GARCIA, Marcus. **Visual Basic .NET: desenvolvendo uma aplicação comercial**. Rio de Janeiro : BRASPORT, 2005. nv, il. (Visual studio 2005 express).

VALOR ECONOMICO. **Mercado de Tecnologia movimentou R\$ 467,8 bilhões no Brasil em 2017**. São Paulo-SP. 2018. Disponível em < <https://www.valor.com.br/empresas/5470331/mercado-de-tecnologia-movimentou-r-4678-bilhoes-no-brasil-em-2017>>. Acesso em 27 ago. 2018.

ANEXO A – Exemplo de XM Gerado pelo Software Cliente

Na figura 9 ve-se um XML gerado a partir da coleta executado pelo serviço cliente do software auditco. É possível visualizar de forma detalhada todas as informações fornecidas pelo *guest* no determinado momento da coleta. Essas informações podem variar de máquina para máquina e de Windows para Windows conforme configuração disponível.

Quadro 11 – Conteúdo de XML gerado pelo software cliente

```
<?xml version="1.0" encoding="utf-8"?>
<AuditCo_Cliente>
  <Coleta>
    <datacoleta>2018-7-1</datacoleta>
    <horacoleta>21:0:15</horacoleta>
    <nmhost>NBBNU006731</nmhost>
    <nmuserlogado>Diego.Pamplona</nmuserlogado>
    <nmcontdomhost>HONDA</nmcontdomhost>
    <uptimehost>0 dias e 5 horas e 20 minutos e 21 segundos</uptimehost>
    <timezonehost>Hora oficial do Brasil</timezonehost>
    <prochost>Intel(R) Core(TM) i5-3230M CPU @ 2.60GHz</prochost>
    <qtdcorhost>4 Cores</qtdcorhost>
    <nmsoatualhost>Microsoft Windows 10 Pro</nmsoatualhost>
    <platsoatualhost>64 Bits</platsoatualhost>
    <lingsoatualhost>pt-BR</lingsoatualhost>
    <qtdmemhost>8GB</qtdmemhost>
    <qtdmemdisphost>3GB</qtdmemdisphost>
    <discosohost>C:\</discosohost>
    <formatdiscosohost>NTFS</formatdiscosohost>
    <tamtotdiscosohost>298GB</tamtotdiscosohost>
    <tamdispdiscosohost>200GB</tamdispdiscosohost>
    <nomeniccabo>Conexão local</nomeniccabo>
    <descniccabo>Broadcom NetXtreme 57xx Gigabit Controller</descniccabo>
    <macaddressniccabo>A41F72F7ECC6</macaddressniccabo>
    <enderipniccabo>169.254.148.84</enderipniccabo>
    <nomenicsemfio>Conexão de Rede sem Fio</nomenicsemfio>
    <descnicsemfio>Adaptador de rede Broadcom 802.11n</descnicsemfio>
    <macaddressnicsemfio>342387088762</macaddressnicsemfio>
```

<enderipnicsemfio>192.168.1.104</enderipnicsemfio>
<placavideo>Intel(R) HD Graphics 4000 com 2GB Dedicado de RAM</placavideo>
<resolucaoatual>1366 X 768</resolucaoatual>
<impressorainst1>Send To OneNote 2016</impressorainst1>
<impressorainst2>PDFCreator</impressorainst2>
<impressorainst3>PDF Architect 4</impressorainst3>
<impressorainst4>Microsoft XPS Document Writer</impressorainst4>
<impressorainst5>Microsoft Print to PDF</impressorainst5>
<impressorainst6>Fax</impressorainst6>
<impressorainst7>EPSON XP-211 214 216 Series</impressorainst7>
<chavewindows>TY4CG-JDJH7-VJ2WF-DY4X9-HCFC6</chavewindows>
<chaveoffice>N/A</chaveoffice>
<instanciasql>NBBNU006731SQLEXPRESS</instanciasql>
<programinstal1>Driver Easy 5.6.2</programinstal1>
<programinstal2>EPSON XP-211 214 216 Series Printer Uninstall</programinstal2>
<programinstal3>Git version 2.16.1.4</programinstal3>
<programinstal4>Microsoft Help Viewer 1.1</programinstal4>
<programinstal5>Microsoft Help Viewer 1.1 Language Pack - PTB</programinstal5>
<programinstal6>System Center Endpoint Protection</programinstal6>
<programinstal7>Microsoft Office 365 ProPlus - pt-br</programinstal7>
<programinstal8>PDFCreator</programinstal8>
<programinstal9>PDF Architect 4 View Module</programinstal9>
<programinstal10>IIS Express Application Compatibility Database for
x64</programinstal10>
<programinstal11>Microsoft ASP.NET Core 2.0.5 Runtime Package Store
(x64)</programinstal11>
<programinstal12>Windows Firewall Configuration Provider</programinstal12>
<programinstal13>IIS 10.0 Express</programinstal13>
<programinstal14>Intel(R) Management Engine Components</programinstal14>
<programinstal15>Microsoft Visual C++ 2010 x64 Redistributable -
10.0.40219</programinstal15>
<programinstal16>Microsoft SQL Server 2014 Management Objects
(x64)</programinstal16>
<programinstal17>Microsoft ODBC Driver 11 for SQL Server</programinstal17>
<programinstal18>Microsoft SQL Server 2016 Management Objects
(x64)</programinstal18>
<programinstal19>Microsoft Visual C++ 2017 x64 Minimum Runtime -

14.13.26020</programinstal19>

<programinstal20>Intel(R) Management Engine Components</programinstal20>

<programinstal21>Microsoft .NET Framework 4.5.2</programinstal21>

<programinstal22>Microsoft Visual Studio 2015 Update 3 Diagnostic Tools - amd64</programinstal22>

<programinstal23>Active Directory Authentication Library for SQL Server</programinstal23>

<programinstal24>Microsoft .NET Core Host - 2.0.5 (x64)</programinstal24>

<programinstal25>Dell SupportAssist</programinstal25>

<programinstal26>Microsoft Build Tools Language Resources 14.0 (amd64)</programinstal26>

<programinstal27>Microsoft Visual C++ 2012 x64 Additional Runtime - 11.0.61030</programinstal27>

<programinstal28>Intel® Trusted Connect Service Client</programinstal28>

<programinstal29>Java(TM) SE Development Kit 10.0.1 (64-bit)</programinstal29>

<programinstal30>Microsoft Visual Studio 2015 Performance Collection Tools - ENU</programinstal30>

<programinstal31>Serv. de Compilador Transact-SQL do Microsoft SQL Server 2014</programinstal31>

<programinstal32>Visual Studio 2010 Prerequisites - English</programinstal32>

<programinstal33>Microsoft .NET Core SDK - 2.1.100 (x64)</programinstal33>

<programinstal34>Microsoft Visual C++ 2008 Redistributable - x64 9.0.30729.4148</programinstal34>

<programinstal35>Microsoft Visual Studio Team Foundation Server 2015 Update 3 CTP1 Storyboarding (x64)</programinstal35>

<programinstal36>Windows Software Development Kit DirectX x64 Remote</programinstal36>

<programinstal37>Microsoft Sync Framework Runtime v1.0 (x64)</programinstal37>

<programinstal38>UpdateAssistant</programinstal38>

<programinstal39>Microsoft Visual Studio 2015 VsGraphics Helper Dependencies</programinstal39>

<programinstal40>Intel(R) Management Engine Components</programinstal40>

<programinstal41>PeaZip 5.1.1 (WIN64)</programinstal41>

<programinstal42>Gravador VSS da Microsoft para SQL Server 2014</programinstal42>

<programinstal43>Microsoft Visual C++ 2008 Redistributable - x64 9.0.30729.6161</programinstal43>

<programinstal44>Visual C++ IDE x64 Package</programinstal44>

<programinstal45>Microsoft Visual Studio 2015 Diagnostic Tools -

amd64</programinstal45>

<programinstal46>Microsoft Visual Studio Team Foundation Server 2015 Update 3 CTP1 Office Integration (x64)</programinstal46>

<programinstal47>Microsoft Policy Platform</programinstal47>

<programinstal48>Configuration Manager Client</programinstal48>

<programinstal49>PDF Architect 4 Create Module</programinstal49>

<programinstal50>Microsoft .NET Core Runtime - 2.0.5 (x64)</programinstal50>

<programinstal51>Microsoft SQL Server Compact 4.0 SP1 x64 ENU</programinstal51>

<programinstal52>Microsoft Build Tools 14.0 (amd64)</programinstal52>

<programinstal53>Microsoft Sync Services for ADO.NET v2.0 (x64)</programinstal53>

<programinstal54>Visual Studio 2015 Prerequisites - ENU Language Pack</programinstal54>

<programinstal55>Microsoft Visual Studio 2015 Update 3 IntelliTrace (x64)</programinstal55>

<programinstal56>Microsoft Silverlight</programinstal56>

<programinstal57>Intel(R) Management Engine Components</programinstal57>

<programinstal58>Office 16 Click-to-Run Licensing Component</programinstal58>

<programinstal59>Office 16 Click-to-Run Extensibility Component</programinstal59>

<programinstal60>Office 16 Click-to-Run Localization Component</programinstal60>

<programinstal61>Microsoft Web Deploy 3.6</programinstal61>

<programinstal62>Microsoft Application Error Reporting</programinstal62>

<programinstal63>Microsoft System CLR Types for SQL Server 2016</programinstal63>

<programinstal64>Windows Software Development Kit for Windows Store Apps DirectX x64 Remote</programinstal64>

<programinstal65>Microsoft SQL Server 2014 Transact-SQL ScriptDom</programinstal65>

<programinstal66>Microsoft SQL Server 2012 Command Line Utilities</programinstal66>

<programinstal67>Dell Touchpad</programinstal67>

<programinstal68>Revo Uninstaller 2.0.5</programinstal68>

<programinstal69>Microsoft Forefront Endpoint Protection 2010 Server Management</programinstal69>

<programinstal70>Microsoft SQL Server 2014 Express LocalDB </programinstal70>

<programinstal71>Update for Windows 10 for x64-based Systems (KB4023057)</programinstal71>

<programinstal72>IIS Express Application Compatibility Database for x86</programinstal72>

```

<programinstal73>Microsoft Visual C++ 2005 Redistributable (x64)</programinstal73>
<programinstal74>Microsoft Visual Studio Team Foundation Server 2015 Update 3
CTP1 Storyboarding Language Pack (x64) - ENU</programinstal74>
<programinstal75>Microsoft .NET Framework 4.5.1 (PTB)</programinstal75>
<programinstal76>Microsoft Visual Studio 2015 Update 3 Diagnostic Tools -
amd64</programinstal76>
<programinstal77>PDF Architect 4 Edit Module</programinstal77>
<programinstal78>Microsoft .NET Version Manager (x64) 1.0.0-
beta5</programinstal78>
<programinstal79>Microsoft Visual C++ 2017 x64 Additional Runtime -
14.13.26020</programinstal79>
<programinstal80>Microsoft Help Viewer 1.1 Language Pack - PTB</programinstal80>
<programinstal81>Microsoft Visual C++ 2012 x64 Minimum Runtime -
11.0.61030</programinstal81>
<programinstal82>Microsoft SQL Server 2016 T-SQL ScriptDom </programinstal82>
<programinstal83>Java 10.0.1 (64-bit)</programinstal83>
<programinstal84>Microsoft Security Client</programinstal84>
<programinstal85>Microsoft SQL Server 2012 Native Client </programinstal85>
<programinstal86>Microsoft .NET Core Host FX Resolver - 2.0.5
(x64)</programinstal86>
<programinstal87>Visual Studio 2015 Prerequisites</programinstal87>
<programinstal88>Managed Windows Defender</programinstal88>
<programinstal89>Microsoft SQL Server 2016 LocalDB </programinstal89>
<programinstal90>Microsoft Help Viewer 1.1</programinstal90>
<programinstal91>Microsoft Visual Studio Team Foundation Server 2015 Update 3
CTP1 Office Integration Language Pack (x64) - ENU</programinstal91>
<programinstal92>Intel(R) ME UninstallLegacy</programinstal92>
<programinstal93>Microsoft System CLR Types for SQL Server
2014</programinstal93>
<programinstal94>Microsoft Visual Studio 2015 Performance Collection
Tools</programinstal94>
</Coleta>
</AuditCo_Cliente>

```