

UNIVERSIDADE FEEVALE

WILLIAM HUMBERTO DO COUTO BECHER

ANÁLISE COMPARATIVA DE BANCOS DE DADOS DE
IMAGENS MÉDICAS OPEN SOURCE

Novo Hamburgo
2017

WILLIAM HUMBERTO DO COUTO BECHER

ANÁLISE COMPARATIVA DE BANCOS DE DADOS DE
IMAGENS MÉDICAS OPEN SOURCE

Trabalho de Conclusão de Curso
apresentado como requisito
parcial à obtenção do grau de
Bacharel em Ciência da Computação
pela Universidade Feevale.

Orientador: Ricardo Ferreira de Oliveira

Novo Hamburgo
2017



INSTITUTO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
UNIVERSIDADE FEEVALE

Trabalho de Conclusão de Curso de Ciência da Computação, intitulado: ***ANÁLISE COMPARATIVA DE BANCOS DE DADOS DE IMAGENS MÉDICAS OPEN SOURCE*** de autoria de William Humberto do Couto Becher, aprovada pela banca examinadora constituída pelos seguintes professores:

Prof. Me. Ricardo Ferreira de Oliveira
Universidade Feevale

Prof. Me. Edvar Bergmann Araujo
Universidade Feevale

Prof. Dra. Marta Rosecler Bez
Universidade Feevale

Coordenador(a) do Departamento Instituto de Ciências Exatas e Tecnológicas
Prof. Dr. Juliano Varella de Carvalho
ICET/FEEVALE

Novo Hamburgo, 11 de Novembro de 2017

RESUMO

Com a modernização proporcionada pelos computadores, e a agilidade necessária para identificação dos sintomas, de modo a acelerar o tratamento dos pacientes, a utilização de bancos de dados de imagens médicas vem sendo adotada por cada vez mais hospitais e clínicas de diagnóstico. Os softwares utilizados para o armazenamento destas imagens médicas, são conhecidos como PACS (*Picture Archiving and Communication System*). Existem diversos softwares que desempenham esta função, e este trabalho tem como objetivo analisar alguns PACS *open source* utilizados atualmente, e desenvolver um instrumento de avaliação de modo a auxiliar os profissionais que necessitem utilizar esses softwares a escolherem o que melhor se adapta às suas necessidades.

No decorrer deste trabalho, serão comparados os PACS *open-source*: Osirix/Horos, Orthanc, Dicoogle, Conquest DICOM e dcm4che, com o objetivo de descrever o funcionamento e características presentes em cada um dos softwares. A metodologia utilizada foi a instalação dos softwares e a execução de um roteiro de uso. Foram analisados critérios específicos de uso, e gerais, tais como: instalação, armazenamento de arquivos, utilização do sistema, comunicação entre serviços, visualização de exames e possibilidades de customização.

A análise empírica do autor destaca um dos PACS como a melhor opção de visualização de exames. E outro como melhor software para ser utilizado como servidor, devido à sua API de comunicação WEB. Houveram problemas em analisar alguns dos softwares, devido à dificuldade de instalação e pouca documentação disponível.

Com essas informações, este trabalho poderá ser usado para consultas de empresas que necessitem implementar um sistema de imagens médicas digitais em seu ambiente de trabalho, ou até mesmo para auxiliar estudantes da área da medicina que precisem de um PACS para uso individual.

Palavras-chave: Avaliação de Software. DICOM. Imagens Médicas. PACS.

ABSTRACT

With the modernization provided by computers, and the agility necessary to identify symptoms, in order to accelerate the treatment of patients, the use of medical imaging databases has been increasingly adopted in hospitals and diagnostic clinics. The software used to store these medical images is known as PACS (Picture Archiving and Communication System). There are several softwares that perform this function, and this work aims to analyze some open source PACS currently used, and develop an evaluation tool to help professionals who need to use these softwares to choose the one that best suits their needs .

In this paper, the following open-source PACS will be compared: Horos, Orthanc, Dicoogle, Conquest DICOM e dcm4che, with the objective of describing the functionality and characteristics present in each of the softwares. The methodology used was the installation of the software and the execution of a roadmap of use. Specific and general usage criteria were analyzed, such as: installation, file storage, system utilization, service communication, examination visualization and customization possibilities.

Author emphasizes one of the PACS as the best option for exam screening. And another one the better software to be used as a server, because of its API of communication to the WEB. There have been problems in analyzing some of the software because of the difficulty of installing it and little documentation available.

With this information, this work can be used for company inquiries that need to implement a system of digital images in its work environment, or even to assist medical students needing PACS for individual use

Keywords: Software Evaluation. DICOM. Medical Images. PACS.

LISTA DE FIGURAS

Figura 1	Esquema da arquitetura de rede de um PACS com vários clientes	12
Figura 2	Comandos DICOM suportados pelo Orthanc	18
Figura 3	Modelo de qualidade para qualidade externa e interna de acordo com a ISO/IEC 9126	22
Figura 4	Modelo de qualidade para qualidade em uso de acordo com a ISO/IEC 9126	25
Figura 5	Perguntas a serem realizadas para cada subcaracterística	26
Figura 6	Especificações da máquina de testes	30
Figura 7	<i>Screenshot</i> do site Orthanc Books	34
Figura 8	Página da documentação do Dicoogle	36
Figura 9	Captura de tela com as pastas de arquivos do Horos	39
Figura 10	Estrutura de pastas do Orthanc Server	41
Figura 11	Tela de progresso do indexador do Dicoogle	42
Figura 12	Tela Inicial do Horos	44
Figura 13	Tela Inicial do Orthanc	45
Figura 14	Tela da listagem de exames de um paciente no Orthanc	45
Figura 15	Resultados de busca do Dicoogle - Pacientes	46
Figura 16	Resultados de busca do Dicoogle - Imagens	47
Figura 17	Tela de configuração do <i>Listener</i> DICOM do Horos	48
Figura 18	Tela de configuração de locais DICOM do Horos	49
Figura 19	Resposta de uma <i>query</i> à API do Orthanc Server	50
Figura 20	Tela de configuração dos serviços DICOM do Dicoogle	51
Figura 21	Visualização 3D com renderização de superfície do Horos	52
Figura 22	Visualização 2D do Horos	53
Figura 23	Tela de visualização de exames do Orthanc Web Viewer	54
Figura 24	Tela de visualização de imagens do Dicoogle	55
Figura 25	Gráfico de Commits do projeto Horos no Github	56
Figura 26	Tela de escolha de plugins do Horos	57
Figura 27	Gráfico de Commits do projeto Dicoogle no Github	58

LISTA DE QUADROS

Quadro 1	Comparação básica entre os PACS a serem analisados	20
Quadro 2	Seções do arquivo de configuração do Orthanc	38
Quadro 3	Distribuição de exames e quantidade de arquivos	43
Quadro 4	Opções disponíveis no visualizador web do Orthanc	54
Quadro 5	Síntese das características analisadas nos PACS	60

LISTA DE ABREVIATURAS E SIGLAS

SIGLA - NOME COMPLETO

API - *Application Programming Interface*

DICOM - *Digital Imaging and Communications in Medicine*

IEC - *International Electrotechnical Commission*

ISO - *International Organization for Standardization*

LAN - *Local Area Network*

NBR - Norma Brasileira

PACS - *Picture Archiving and Communication System*

REST - *Representational State Transfer*

UUID - *Universally Unique Identifier*

WADO - *Web Access to DICOM Objects*

WAN - *Wide Area Network*

SUMÁRIO

1	Introdução	11
2	Os Sistemas PACS	14
2.1	Exames médicos através de imagens	14
2.2	PACS: Visão Geral	15
2.3	O Núcleo de Funcionamento de um PACS	15
2.4	As vantagens de um PACS	16
2.4.1	Segurança da Informação	16
2.4.2	Visualização simultânea	16
2.4.3	Organização e catalogação dos exames	17
2.4.4	Pós-processamento e melhoramento	17
2.4.5	Economia	17
2.5	PACS a serem analisados	18
2.5.1	Horos	18
2.5.2	Orthanc	18
2.5.3	dcm4che	19
2.5.4	Conquest DICOM	19
2.5.5	Dicoogle	19
3	Qualidade de Software - Aspectos gerais	21
3.1	Qualidade Interna e Externa	22
3.1.1	Funcionalidade	23
3.1.2	Confiabilidade	23
3.1.3	Usabilidade	23
3.1.4	Eficiência	24
3.1.5	Manutenibilidade	24
3.1.6	Portabilidade	24
3.2	Qualidade em Uso	25
4	Metodologia	26
4.1	Instrumento de Avaliação	26
4.1.1	Ambiente de Validação	30
4.1.2	<i>Workflow</i> básico de utilização do software	31
5	Análise dos Softwares	33
5.1	Documentação	33
5.1.1	Horos	33
5.1.2	Documentação do Orthanc	34

	10
5.1.3 Documentação do Dicoogle	35
5.2 Instalação e configuração dos softwares	36
5.2.1 Horos	36
5.2.2 Instalação do Orthanc	37
5.2.3 Dicoogle	37
5.3 Configurações de Armazenamento	38
5.3.1 Horos	39
5.3.2 Orthanc	40
5.3.3 Dicoogle	41
5.4 Roteiro de Utilização	43
5.4.1 Horos	43
5.4.2 Orthanc	44
5.4.3 Dicoogle	46
5.5 Métodos de Comunicação	47
5.5.1 Horos	47
5.5.2 Orthanc	49
5.5.3 Dicoogle	50
5.6 Visualização de Exames	50
5.6.1 Horos	51
5.6.2 Orthanc	53
5.6.3 Dicoogle	55
5.7 Possibilidades de Customização	55
5.7.1 Horos	55
5.7.2 Orthanc	57
5.7.3 Dicoogle	58
5.8 Dificuldades Encontradas	59
5.8.1 Dificuldades com o dcm4chee	59
5.8.2 Dificuldades com o Conquest	59
6 Conclusão	61
6.1 Trabalhos Futuros	62
REFERÊNCIAS BIBLIOGRÁFICAS	63

1 Introdução

Dreyer e Hirschorn (2006) escrevem em seu livro, *PACS: A Guide to the Digital Revolution*, que os maiores departamentos de radiologia tornaram-se digitais, e os centros de diagnóstico por imagem menores não ficaram para trás. Segundo os autores, imprimir exames de imagem para interpretá-los é como imprimir um e-mail para poder lê-lo. E os benefícios da utilização de PACS (*Picture Archiving and Communication System*) são claros. Segundos após um exame ser realizado e a imagem transferida para o sistema, ela pode ser vista pelo radiologista e por qualquer médico que precise analisá-la simultaneamente. Tomografias Computadorizadas com milhares de imagens estão tornando-se comuns e são muito melhor gerenciadas em sistemas de arquivos digitais.

De acordo com Seixas (2005), a demanda dos hospitais por um diagnóstico de exames de imagens digitais de forma rápida e precisa, somada aos avanços computacionais e de processamento de imagens, fizeram surgir novas frentes de pesquisa relacionadas ao diagnóstico radiológico auxiliado por computador.

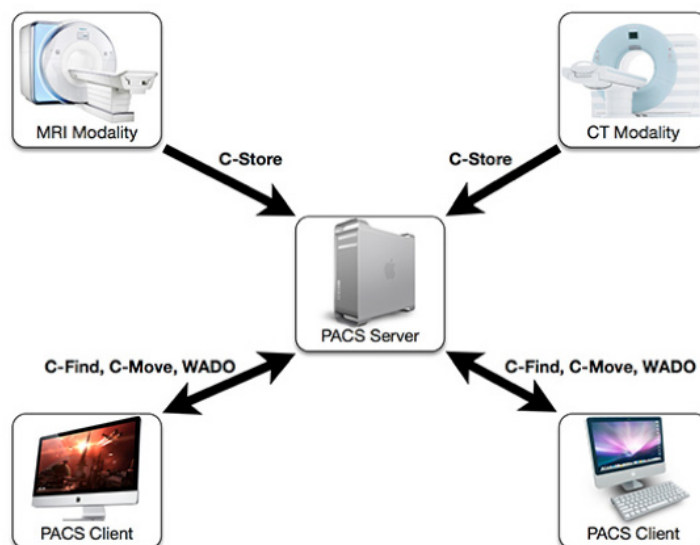
A implantação de um serviço de radiologia “*filmless*” tem resultado em grande número de avanços operacionais, incluindo melhoria no gerenciamento das imagens e leitura mais rápida, possibilitando acessos quase que em tempo real, eliminação de certo número de passos no processo de disponibilização das imagens, eliminação de exames perdidos e melhoria na produtividade do trabalho em grupo. (CARITÁ et al., 2004)

Segundo a definição da *NEMA (American National Association of Electric Machines)*, um PACS deve oferecer visualização de imagens em estações de diagnóstico remotas; armazenamento de dados em meios magnéticos ou ópticos para recuperação em curto ou longo prazo; comunicação utilizando redes locais (*Local Area Network, LAN*) ou expandidas (*Wide Area Network, WAN*), ou outros serviços públicos de telecomunicação; sistemas com interfaces por modalidade e conexões para serviços de saúde e informações departamentais que ofereçam uma solução integrada para o usuário final. (DE AZEVEDO-MARQUES and SALOMÃO, 2009)

É importante que a implantação do sistema PACS em um hospital ou clínica atenda seus requisitos, pois o desempenho do sistema exerce forte influência no atendimento aos pacientes. (SEIXAS, 2005)

Segundo Ratib (2016), há uma tendência mundial no desenvolvimento de softwares open-source na medicina, que provêm uma alternativa robusta e confiável para soluções de visualização e análise de imagens médicas. Esses softwares, implementam o padrão DICOM (*Digital Imaging and Communications in Medicine*), que segundo Azevedo-Marques, Salomão (2009) é o padrão global para transferência de imagens radiológicas e outras informações médicas entre computadores. Para um software ser con-

Figura 1: Esquema da arquitetura de rede de um PACS com vários clientes



Fonte: (OSIRIX, 2017)

siderado um PACS, ele deve seguir esse padrão DICOM, e permitir o uso de comandos DICOM para interação entre os clientes e o servidor. Como pode-se perceber na Figura 1, a arquitetura de rede de um PACS permite a comunicação e transferência de imagens médicas entre vários dispositivos diferentes.

A qualidade do software utilizado é importante para qualquer empresa, pois a falta de qualidade pode gerar diversos prejuízos, e na área da saúde esse ponto é ainda mais importante, visto que um mal funcionamento pode acarretar na demora no atendimento de um paciente. A ISO/IEC 9126 (NBR13596) (ISO/IEC, 2001) propõe um modelo de avaliação de software com seis categorias básicas a serem analisadas: funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade.

Pelo acima exposto, este trabalho propõe uma comparação entre alguns dos PACS Open-Source existentes no mercado, de maneira a facilitar o processo de escolha do software que melhor se adapta as necessidades das instituições. Considerando isto, serão realizados testes de *benchmark*, testes de usabilidade e comparações das características de cada software, avaliando as categorias da ISO/IEC 9126 (NBR13596) (ISO/IEC, 2001) citadas anteriormente. Pretende-se comparar os seguintes PACS: OsiriX/Horos, Orthanc, Conquest, dcm4che e Dicoogle.

O presente trabalho é dividido em seis capítulos, sendo eles: Introdução; Os Sistemas PACS - onde é abordada uma visão geral sobre os exames médicos de imagens digitais, o funcionamento básico de um PACS, e uma breve apresentação sobre os softwares a serem analisados; Qualidade de Software - capítulo que apresenta o que é a qualidade de software segundo a ISO e a norma brasileira; Metodologia - onde é apresen-

tada a forma de análise e critérios utilizados durante o trabalho; Análise dos Softwares - o detalhamento da análise dos PACS; e a Conclusão, que apresenta os resultados obtidos durante a execução deste trabalho.

A relevância desta pesquisa contribui, diretamente, para ser utilizada em estudos futuros com relação a PACS *open-source*, e serve de base para outros trabalhos que venham a abordar este tema, sendo que este trabalho sintetiza diversas características de funcionamento de alguns PACS *open-source*.

2 Os Sistemas PACS

Este capítulo apresenta uma visão geral do funcionamento de um PACS. Nele serão apresentados os exames médicos através de imagens, uma visão geral dos sistemas PACS, o Núcleo de Funcionamento de um PACS, suas vantagens e características.

2.1 Exames médicos através de imagens

O Raio-X foi descoberto em 1895 pelo professor Wilhem Conrad Roentgen (1845-1923), um professor da Universidade de Wuerzburg na Alemanha, enquanto trabalhava com um tubo de raios catódicos em seu laboratório. Ele acabou descobrindo que o raio recentemente descoberto podia passar através da maioria das substâncias, evidenciando as sombras de objetos sólidos. Roentgen também notou que o raio atravessava tecidos humanos, mas não os ossos e objetos de metal. De 1895 até agora, a radiografia mudou pouco. Nós ainda capturamos uma imagem da sombra de um objeto em filme utilizando procedimentos similares aos de antigamente. Atualmente, entretanto, consegue-se gerar imagens com uma qualidade superior e maior definição devido a variedade e qualidade dos filmes disponíveis no mercado hoje em dia. O processamento de filmes evoluiu para um estado automatizado. A eletrônica e os computadores permitem capturar essas imagens digitalmente, e armazená-las em servidores em qualquer lugar do mundo, sem deterioração devido ao tempo. Esses avanços tecnológicos permitiram a indústria a produzir raios-x com qualidade superior, um custo menor e uma rapidez maior. (NDT RESOURCE CENTER, 2013)

Silva (2015) afirma que os métodos avançados de diagnóstico por imagem vêm se firmando, na sua aplicação clínica, tornando-se essenciais para o planejamento de casos, auxiliando no diagnóstico e permitindo uma avaliação diferenciada e detalhada da anatomia envolvida, e patologias. Estes exames permitem economia de tempo do profissional em procedimentos mais complexos, e até mesmo nas consultas de rotina.

A aplicação de sistemas de informação para gerenciamento de imagens e informações clínicas começou a ser estudada de forma mais efetiva no final da década de 1980, quando os processos de aquisição digital começaram a ser utilizados em maior escala nos hospitais. (DE AZEVEDO-MARQUES and SALOMÃO, 2009)

De acordo com Valente (2016), o conceito de PACS começou a ser largamente aceito duas décadas atrás. As vantagens destes sistemas são bem entendidos pela comunidade médica, e os obstáculos iniciais, como as tecnologias de visualização, armazenamento e transmissão foram superados. Para as instituições menores, a questão do uso de um PACS não é mais formulada em termos de "se", mas em termos de "quando" e "como".

2.2 PACS: Visão Geral

O desenvolvimento do uso da informação em formato digital criou a necessidade de se estabelecer uma estrutura computacional que possibilitasse a troca de dados de imagens de forma consistente e automática dentro do ambiente hospitalar. Em resposta a essa necessidade, surgiu o conceito de PACS (do inglês, *Picture Archiving and Communication System*). (DE AZEVEDO-MARQUES and SALOMÃO, 2009) A gama de imagens que podem ser digitalizadas em um ambiente hospitalar, é grande, segundo Guimarães (2002), as imagens mais comuns a serem estudadas são os Raios-X, que compreende aproximadamente 70% do total. O restante das imagens produzidas em exames são imagens de radiografia computadorizada (CR), tomografia computadorizada (CT), ressonância magnética (MRI), ultra-sonografia, medicina nuclear (NMI), angiografia de subextração digital (DAS), entre outras, que ocupam os outros 30% do volume.

Até meados da década de 1980, a única forma de aquisição de uma radiografia era por meio de filme. De forma semelhante, quando outras modalidades, como ultrassom, tomografia computadorizada e ressonância magnética surgiram, ainda era necessário imprimir as imagens em filme para visualização. Com o surgimento do PACS, percebeu-se a possibilidade da utilização de monitores acoplados a computadores para visualização de imagens. Os primeiros monitores padrão CRT não forneciam a qualidade necessária em comparação à de um filme radiológico e, portanto, os fabricantes tiveram de produzir monitores específicos para a área médica, com fósforos especiais para atingir níveis maiores de luminância e contraste e menor distorção espacial e latência. Mais tarde, os monitores LCD foram adotados devido ao seu baixo custo, durabilidade e qualidade de exibição, inclusive de imagens coloridas. (DE AZEVEDO-MARQUES and SALOMÃO, 2009)

2.3 O Núcleo de Funcionamento de um PACS

Uma vez que as imagens forem adquiridas, elas precisam ser gerenciadas apropriadamente para garantir que o armazenamento, a recuperação e a entrega ocorram sem erros. O PACS ainda deve garantir que as imagens sejam armazenadas utilizando armazenamento de longo prazo que fiquem de acordo com as obrigações legais para retenção de imagens. Adicionalmente, elas precisam ser entregues para interpretação em tempo hábil. Estes requisitos são satisfeitos pelo núcleo do PACS. (DREYER et al., 2006)

O núcleo de funcionamento é composto dos seguintes componentes:

- Sistema de Banco de Dados (Ex., Oracle, MS-SQL, Sybase)
- Sistema de Arquivos (Ex., RAID, Jukebox)

- Controle de fluxo (gerenciador de imagens)
- Interface RIS (Sistema de Informações Radiológicas)

2.4 As vantagens de um PACS

Apesar de possuir diversas vantagens teóricas do uso de PACS, diversos departamentos que fizeram a transição para operações *filmless* descobriram que embora estivessem reduzindo custos com a não utilização de filmes, e proporcionando acesso melhorado aos exames para os clínicos, eles não estavam alcançando todas as vantagens que os PACS podem oferecer. (SIEGEL and REINER, 2002)

De acordo com Strickland (2000), com a troca dos exames convencionais em filme, pela utilização de uma aquisição de imagens *filmless*, pode-se listar diversas vantagens do gerenciamento digital dos exames, conforme descritas nas sub-seções a seguir:

2.4.1 Segurança da Informação

Uma vez que uma imagem é inserida em um PACS, ela não pode ser perdida, roubada, ou arquivada por engano. Muitos hospitais reportam que 20% dos filmes são perdidos, gerando um problema sério. Além disso, as imagens estão, quase sempre disponíveis rapidamente quando é utilizado um sistema de arquivamento digital de imagens, então dificilmente uma consulta precisa ser cancelada, ou uma decisão médica precisa ser adiada, e pouco tempo é perdido por médicos ou clínicas de aquisição de imagens. As imagens ficam disponíveis dia e noite, dentro e fora do hospital, dependendo do PACS utilizado. (STRICKLAND, 2000)

2.4.2 Visualização simultânea

Os diversos terminais de acesso aos PACS em todo o hospital permitem a visualização simultânea da mesma imagem, em locais diferentes, enquanto que o filme convencional pode existir fisicamente em um só lugar e momento. Isso significa, por exemplo, que um médico no departamento de acidentes e emergências pode discutir as imagens de um paciente com o radiologista, com os dois profissionais visualizando as imagens sem que nenhum dos dois tenha deixado seu departamento. Da mesma forma, no momento em que um paciente retornar para o ambulatório após ser enviado para um exame radiológico urgente, as imagens já estarão disponíveis no PACS para exibição pelo médico solicitante. (STRICKLAND, 2000)

2.4.3 Organização e catalogação dos exames

O banco de dados do PACS garante que todas as imagens sejam agrupadas automaticamente no exame correto, são cronologicamente ordenadas, corretamente orientadas e rotuladas, e podem ser facilmente recuperadas usando uma variedade de critérios (por exemplo, nome do paciente, número de hospital, data, clínico de referência, etc.). Todos os estudos de imagem de um paciente estão imediatamente disponíveis no PACS, que incentiva a revisão de exames anteriores e comparações. (STRICKLAND, 2000)

2.4.4 Pós-processamento e melhoramento

Trabalhar com imagens em monitores permite que a gama completa de ferramentas computacionais sejam utilizadas para manipular e pós-processar as imagens. A alteração do nível do contraste permite que tecido macio e estruturas ósseas sejam bem vistos em uma única visualização. Por exemplo, muitas vezes permite que o lóbulo inferior esquerdo seja avaliado por trás da silhueta cardíaca esquerda, enquanto que esta informação não está disponível em uma radiografia de tórax tradicional em filme. Existe, portanto, um aumento na quantidade de informações que podem ser extraídas de uma imagem, o que é particularmente notável para a radiografia simples. Isso também é em parte o resultado do dispositivo de aquisição de placas de fósforo foto-estimuláveis geralmente usadas para adquirir essas imagens em formato digital. Essas placas de fósforo possuem uma faixa dinâmica maior do que a combinação convencional de tela-filme que leva a uma melhor visualização simultânea de estruturas de radiodensidade diferentes. (STRICKLAND, 2000)

2.4.5 Economia

O uso de PACS permite economias diretas do menor gasto em filmes, pacotes de filmes, produtos químicos para processamento de filmes, economia com salários do pessoal de técnicos da sala escura e funcionários de arquivo de filmes e a redistribuição do espaço anteriormente utilizado para armazenamento dos exames impressos. Embora esse corte de custos seja notável, existem outros custos que são incluídos no processo, precisa-se agora de gerentes de tecnologia da informação, porém com um salário um pouco mais alto que os outros técnicos. O objetivo ao iniciar o uso de PACS em seu ambiente é pelo menos ter um custo neutro em relação à radiologia tradicional. Se a economia vier a acontecer, isto é um bônus, visto que se obtém diversas vantagens em outras áreas, como o controle dos dados e o aumento da eficiência que a clínica obtém. (STRICKLAND, 2000)

2.5 PACS a serem analisados

Esta seção detalha os PACS que foram analisados neste trabalho.

São apresentados os PACS, com uma breve descrição, e no final deste capítulo, pode se ver o Quadro 1, com uma comparação entre algumas características dos softwares.

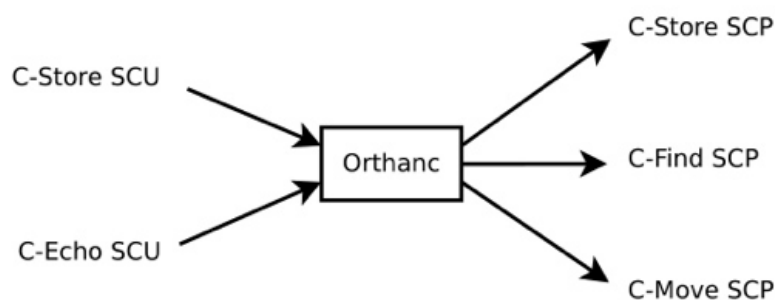
2.5.1 Horos

O OsiriX, software que serviu de base para o Horos (Rosset, 2004), é um software gratuito, distribuído sob o acordo de licença open-source GNU. Segundo Rosset (2004), é um software que foi projetado para visualizar e interpretar grandes conjuntos de imagens médicas multidimensionais. Ele se beneficia da capacidade do padrão OpenGL para mostrar gráficos de maneira muito rápida e otimizada, já que é um padrão amplamente usado pela indústria de games, e consegue fazer uso de qualquer placa de vídeo com aceleradora 3D. Segundo o site oficial (Horos, 2017) Horos, é um software baseado no código fonte do OsiriX (OsiriX, 2017), e pode ser considerado uma versão premium do OsiriX, visto que oferece todas as funcionalidades disponíveis na versão paga deste.

2.5.2 Orthanc

O Orthanc (Jodogne, 2013) é um leve, porém poderoso software de armazenamento DICOM para a área da saúde e pesquisa médica. Múltiplas instâncias do Orthanc podem ser implantadas na rede do hospital, ou até mesmo no mesmo computador, o que simplifica a interconexão entre as diversas aplicações DICOM e facilita a gestão de imagens médicas. De acordo com seu site oficial (Orthanc, 2017), o Orthanc é capaz de transformar qualquer computador em um servidor DICOM, ou em outras palavras, em um mini-PACS. Na Figura 2 podem ser vistos os comandos DICOM suportados pelo Orthanc.

Figura 2: Comandos DICOM suportados pelo Orthanc



Fonte: (ORTHANC, 2017)

2.5.3 dcm4che

O dcm4che (dcm4che, 2017) é um conjunto de aplicações open source e utilitários usados na área da saúde, desenvolvidas em Java, para auxiliar na utilização de imagens médicas. A base desse conjunto de ferramentas foi desenvolvida por Gunter Zeilinger no ano 2000. Em 2006, ele reestruturou as ferramentas, focando as melhorias em performance, uso de memória, flexibilidade e simplicidade de uso.

2.5.4 Conquest DICOM

O Conquest DICOM (Conquest, 2017), é um servidor de imagens médicas criado por Marcel van Herk e Lambert Zijp, no Instituto do Câncer da Holanda. Foi utilizado como base o código fonte em C++ de domínio público do UCDMC DICOM, de Mark Oskin. O Conquest funciona como servidor das imagens, além de poder ser utilizado como visualizador, entre outras funcionalidades. O servidor pode ser instalado com diversos bancos de dados, ODBC, DbaseIII, MySQL, Postgres, Microsoft SQL Server e SQLite.

2.5.5 Dicoogle

De acordo com Costa (2011), Dicoogle é um PACS que utiliza uma indexação baseada em documento, e protocolos *peer-to-peer* (P2P). Substituindo o tradicional banco de dados relacional por uma organização documental, que permite coletar e indexar dados de repositórios baseados em arquivos, o que permite realizar consultar de texto livre. E como resultado dessa estratégia, é que mais informação pode ser extraída de repositórios de imagens digitais, quando comparados com os atuais serviços DICOM que não se utilizam desta técnica. Em seu site oficial, é informado que o Dicoogle (Dicoogle, 2017) oferece um *set* completo de APIs que são capazes de fornecer suporte para os ambientes mais variados, com suporte a diversos dispositivos de acesso diferentes.

Quadro 1: Comparação básica entre os PACS a serem analisados

Software	OsiriX / Horos	Orthanc	dcm4che	Conquest	Dicoogle
Linguagem Utilizada	Objective-C	C / C++	Java, XML	Borland Delphi / C++ (base)	Java
Sistema Operacional Suportado	Mac OSX	Windows / Mac OSX	Windows / Linux / MacOSX	Windows / Linux	Windows / Linux / Mac
Banco de Dados	Próprio / File-system	PostgreSQL	MySQL	Dbase III, ODBC, MicrosoftSQL Server, MySQL, SQLite	Próprio / File-system

Fonte: Próprio Autor

3 Qualidade de Software - Aspectos gerais

Este capítulo apresenta uma fundamentação teórica sobre a qualidade de software. Nele será explanado sobre os aspectos gerais da qualidade de software, a importância dela na construção do software, e uma visão sobre a ISO/IEC 9126 (NBR13593), que descreve um modelo de qualidade de software.

Durante o curso de qualquer treinamento profissional, sempre é exposta a importância da qualidade. Nenhuma outra área profissional exige tanta qualidade quanto a medicina, onde um lapso pode afetar cada parte envolvida com a cadeia do diagnóstico médico através de imagens, desde o agendamento de um exame até as decisões médicas realizadas pelo médico de referência de acordo com o resultado das imagens. Erros nessa área podem ocasionar resultados adversos menores, como tempos de espera mais longos para realização de exames e obtenção de resultados e algumas adversidades maiores, como diagnósticos errados. As rigorosas medidas da garantia de qualidade, originalmente emprestadas de modelos industriais, agora são amplamente reconhecidas como algo importante, mas ainda não são implementadas em todas as empresas. A garantia de qualidade é frequentemente citada como a melhor maneira de garantir um tratamento adequado ao paciente e diminuir possíveis consequências médicas. (DREYER et al., 2006)

No começo do século XX, da década de 20 até o final da Segunda Guerra Mundial, qualidade era definida pelo controle do produto final, ou seja, significava apenas assegurar que o produto defeituoso não chegasse ao usuário. Na década de 60, qualidade passou a ser designada através do controle de todas as etapas de produção do produto. Já na década de 70, normas e procedimentos formais começaram a surgir com o intuito de garantir a qualidade e, hoje, o conceito de qualidade está envolvido com a qualidade total no sentido de satisfação do usuário ou cliente e competitividade. (GLADCHEFF, 2001)

A ISO/IEC 9126 (NBR13596) (ISO/IEC, 2001) descreve um modelo de qualidade do produto de software composto de duas partes: a) qualidade interna e qualidade externa e b) qualidade em uso. Essa primeira parte, especifica seis características para a qualidade interna e externa, que são ainda subdivididas em sub-características. Estas sub-características são atributos internos do software. E a segunda parte do modelo, especifica quatro características da qualidade em uso, que é, para o usuário, o efeito combinado das seis características de qualidade do produto de software. A ISO/IEC 9126 (NBR13596) (ISO/IEC, 2001) define a qualidade como "a capacidade do produto de software em satisfazer as necessidades implícitas e explícitas quando usado em condições específicas". As necessidades explícitas são aquelas que podem ser expressas na definição de requisitos propostos pelo produtor do software, e as implícitas são aquelas que podem não estar descritas em documentos, porém são necessárias pelo usuário. (GLADCHEFF,

2001)

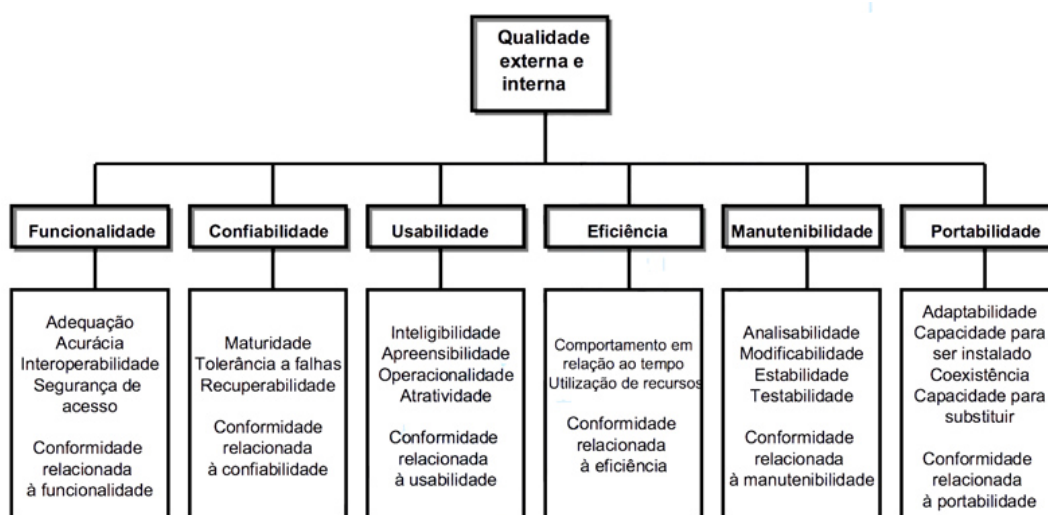
3.1 Qualidade Interna e Externa

De acordo com a ISO/IEC 9126 (NBR13596), o modelo de qualidade interna e externa pode ser categorizado em seis características (funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade, portabilidade) as quais são ainda subdivididas em sub-características, conforme a figura 3.

Qualidade interna são as características do produto de software do ponto de vista interno. Os detalhes da qualidade interna podem ser melhorados durante a implementação do código, revisão, teste. Requisitos de qualidade interna podem ser usados como metas para validação em vários estágios de desenvolvimento. (ISO/IEC, 2001)

Qualidade externa são as características do produto de software do ponto de vista externo. É a qualidade percebida quando o software é executado, normalmente testado, medido e avaliado em um ambiente simulado, com dados simulados e utilizando métricas externas. Durante esses testes, convém que a maioria dos defeitos seja descoberta e eliminada. (ISO/IEC, 2001)

Figura 3: Modelo de qualidade para qualidade externa e interna de acordo com a ISO/IEC 9126



Fonte: Próprio Autor, adaptado de (ISO/IEC, 2001)

Emer (2017) define as sub-características do modelo de qualidade interna e externa conforme as próximas seções:

3.1.1 Funcionalidade

Funcionalidade é a capacidade do produto de software de prover funções que atendam às necessidades explícitas e implícitas quando o software estiver sendo utilizado sob condições especificadas. Ela caracteriza efetivamente o que o software faz. As sub-características da funcionalidade:

- Adequação: presença das funções especificadas;
- Acurácia: gerar os resultados precisos ou esperados;
- Interoperabilidade: interagir e interoperar com outros sistemas conforme o esperado;
- Conformidade: observância de padrões, convenções ou regras já definidas;
- Segurança de acesso: prevenir acesso não autorizado;

3.1.2 Confiabilidade

Confiabilidade é a capacidade do produto em manter um nível de desempenho especificado quando utilizado em condições especificadas. As sub-características da confiabilidade:

- Maturidade: baixa frequência de falhas
- Tolerância a falhas: manter determinados níveis de desempenho mesmo na presença de problemas
- Recuperabilidade: reestabelecer o nível de desempenho desejado e recuperar dados no caso de falhas

3.1.3 Usabilidade

Usabilidade é a capacidade do produto em ser compreendido, aprendido, operado e atraente ao usuário quando usado sob condições especificadas. As sub-características da usabilidade:

- Inteligibilidade: facilidade do usuário reconhecer como é o funcionamento do produto e sua aplicação
- Apreensibilidade: facilidade do usuário em aprender a utilizar o produto
- Operacionalidade: facilidade do usuário para operar o produto

3.1.4 Eficiência

Eficiência é a capacidade do produto de software de apresentar desempenho apropriado, relativo à quantidade de recursos usados, sob condições específicas. As sub-características da eficiência:

- Comportamento em relação ao tempo: medida do tempo de resposta e de processamento
- Comportamento em relação ao uso de recursos: medida de quantidade de recursos necessários e a duração de seu uso ao executar suas funções

3.1.5 Manutenibilidade

Manutenibilidade é a capacidade do produto de software de permitir que uma modificação especificada seja implementada. As sub-características da manutenibilidade:

- Analisabilidade: medida de esforço necessário para diagnosticar ou localizar deficiências ou falhas
- Modificabilidade: medida de esforço necessário para realizar alterações
- Estabilidade: medida de risco de efeitos inesperados provenientes de modificações
- Testabilidade: medida de esforço necessário para testar o software alterado

3.1.6 Portabilidade

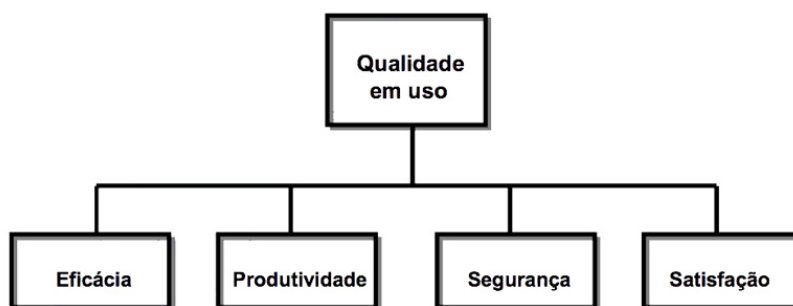
Portabilidade é a capacidade do produto de software de ser transferido de um ambiente para outro. As sub-características da portabilidade:

- Adaptabilidade: medida da facilidade de se adaptar o produto em outros ambientes operacionais
- Facilidade de instalação: medida de esforço necessário para a instalação do produto
- Capacidade para coexistir: medida do nível de conformidade do produto com padrões referentes à portabilidade
- Facilidade para substituir: medida do esforço necessário para que o produto seja usado em substituição a outro

3.2 Qualidade em Uso

A qualidade em uso, de um produto de software é a visão da qualidade percebida sob a perspectiva do usuário. A obtenção da qualidade de uso é dependente da obtenção da qualidade externa, a qual, por sua vez, é dependente da obtenção da qualidade interna. Normalmente são necessárias medidas em todos os três níveis, pois atender aos critérios para medidas internas em geral não é suficiente para garantir o atendimento aos critérios para medidas externas, e atender a esses quesitos não é garantia de que os critérios para qualidade de uso serão atendidos. Na Figura 4 pode-se ver o modelo de qualidade em uso da ISO/IEC 9126.

Figura 4: Modelo de qualidade para qualidade em uso de acordo com a ISO/IEC 9126



Fonte: Próprio Autor, adaptado de (ISO/IEC, 2001)

4 Metodologia

Neste capítulo, são descritos quais critérios serão utilizados neste instrumento de avaliação de software. São apresentadas as perguntas básicas de cada característica de avaliação conforme a ISO/IEC 9126 (NBR13596) e listados os critérios utilizados, com uma breve descrição de cada item. O propósito deste trabalho é avaliar alguns PACS Open Source do mercado, comparando entre eles as características que serão especificadas neste capítulo.

4.1 Instrumento de Avaliação

Para avaliar a qualidade de um produto de software é preciso medir e analisar as características que são importantes para o bom funcionamento do mesmo. Neste trabalho, será desenvolvido um instrumento de avaliação de software PACS, sob a luz de características contidas na ISO/IEC 9126 (NBR13596).

Conforme Bianchi (2017), pode-se sintetizar o entendimento do modelo de qualidade interna e externa de um produto de software ISO/IEC 9126 (NBR13596) através de perguntas simples para cada uma das suas características e sub-características, conforme pode ser visto na figura 5.

Figura 5: Perguntas a serem realizadas para cada subcaracterística

Característica	subcaracterística	Pergunta chave para a subcaracterística
Funcionalidade (Satisfaz as necessidades?)	Adequação	Propõe-se a fazer o que é apropriado?
	Acurácia	Faz o que foi proposto de forma correta?
	Interoperabilidade	Interage com os sistemas especificados?
	Conformidade	Está de acordo com as normas, leis, etc?
Confiabilidade (É imune a falhas?)	Segurança de acesso	Evita acesso não autorizado aos dados?
	Maturidade	Com que frequência apresenta falhas?
	Tolerância a falhas	Ocorrendo falhas como ele reage?
	Recuperabilidade	É capaz de recuperar dados em caso de falhas?
Usabilidade (É fácil de usar?)	Inteligibilidade	É fácil de entender o conceito e a aplicação?
	Apreensibilidade	É fácil de aprender a usar?
	Operacionalidade	É fácil de operar e controlar?
Eficiência (É fácil rápido e "enxuto"?)	Tempo	Qual é o tempo de resposta, a velocidade de execução?
	Recursos	Quanto recurso usa? Durante quanto tempo?
Manutenibilidade (É fácil de modificar?)	Analísabilidade	É fácil de encontrar uma falha quando ocorre?
	Modificabilidade	É fácil modificar e adaptar?
	Estabilidade	Há grande risco quando se faz alterações?
	Testabilidade	É fácil testar quando se faz alterações?
Portabilidade (É fácil de usar em outro ambiente?)	Adaptabilidade	É fácil adaptar em outros ambientes?
	Instabilidade	É fácil instalar em outros ambientes?
	Conformidade	Está de acordo com padrões de portabilidade?
	Substituibilidade	É fácil usar para substituir outro?

Fonte: (BIANCHI, 2017)

Firmino Filho (2013), afirma que para garantir a disponibilidade, é importante o uso de medidas de tolerância a falhas, tais como: detecção de erros, registro de *logs*,

programas de auditoria, redundância dos dados, redundância de hardware, programas de monitoramento (estado da rede, espaço em disco, *status* do banco de dados, utilização do processador e temperatura) e servidores de *backup*. Portanto, as medidas relacionadas ao software, estarão incluídas nos critérios de avaliação propostos por este trabalho.

Com o objetivo de analisar a qualidade em uso dos softwares, os critérios de avaliação que serão utilizados para nortear a pesquisa foram definidos pelo autor deste trabalho, utilizando como base a ISO/IEC 9126 (NBR13596). Com o acima exposto, as métricas a serem avaliadas nos PACS são apresentadas na sequência, levando em consideração as perguntas a serem respondidas, apresentadas na Figura 5, de maneira a sintetizar o modelo de entendimento de qualidade interna e externa de um produto de software.

Critérios a serem analisados

Os critérios enumerados durante a elaboração deste trabalho, para a realização desta avaliação de software, são listados abaixo. Estes critérios foram estabelecidos de acordo com as características pesquisadas na ISO/IEC 9126 (NBR13596).

- **Utilização do padrão DICOM**

É imprescindível que um PACS utilize o padrão DICOM para a comunicação. Será avaliado o suporte à este padrão, e quais categorias estão disponíveis em cada software.

- **Comunicação via LAN**

Será analisado o suporte à comunicação via LAN pelos PACS, e características do funcionamento.

- **Impressão de Exames**

Em alguns casos, é necessário que o exame do paciente seja impresso. Será verificado o suporte à impressão de exames.

- **Diagnóstico Automático**

Será verificado se o sistema de visualização de exames apresenta algum suporte ao diagnóstico automático, facilitando a identificação do quadro clínico do paciente pelo médico.

- **Armazenamento em Banco de Dados**

Neste critério, é preciso verificar quais os bancos de dados suportados pelos PACS.

- **Visualização de Imagens Locais e via LAN**

O objetivo de ter um PACS instalado é justamente não precisar imprimir os exames em filme, ou papel, evitando o desperdício e agilizando o acesso a informação. Para

tanto, serão realizados testes de acesso às imagens, localmente na máquina em que o PACS está instalado, e via LAN, simulando um ambiente de produção.

- **Compartilhamento de imagens em outras mídias (Suporte a *Backup*)**

O *backup* é uma função imprescindível para garantir a segurança das informações. Será averiguado quais os métodos de *backup* disponíveis.

- **Clareza nas mensagens**

Neste critério, serão verificadas as mensagens do software ao usuário. Para constatar se elas estão claras, ou são vagas para o entendimento dos usuários.

- **Estabilidade**

Um software de qualidade não deve apresentar erros sem explicação, e o uso deve ser estável, sem fechamentos indesejáveis e perda de dados. Será feito um levantamento dos erros encontrados no software durante os períodos de testes, e será testado se o software emprega alguma técnica de detecção e prevenção de falhas.

- **Idiomas Suportados**

Será analisado quais idiomas são suportados pela aplicação, e caso não haja a tradução no idioma necessário, se o software permite que seja criada uma tradução pelo próprio usuário.

- **Sistemas Operacionais Suportados**

É importante para o bom funcionamento do sistema em uma rede de computadores, que o software funcione em mais de um sistema operacional, permitindo a heterogeneidade dos dispositivos conectados a esta rede.

- **Análise da GUI do Sistema**

Neste critério, será elaborada uma análise da interface GUI do sistema, com o objetivo de identificar dificuldades que o usuário possa ter, barreiras que possam comprometer a interação, utilizando o conhecimento empírico do autor deste trabalho.

- **Disponibiliza Interface WEB**

A interface WEB é fator importante para garantir o bom funcionamento do PACS em diversos sistemas operacionais. Para o bom funcionamento de um *WADO Server (Web Access to DICOM Objects)*, será verificada a disponibilização dessa função, e testada em sistemas operacionais Linux, Windows e MacOS, bem como browsers diferentes, tais como: *Microsoft Edge, Google Chrome, Opera e Safari*.

- **Documentação (*Online* ou *offline*)**

A documentação do software é importante, seja para a consulta do usuário do

sistema, quanto da equipe técnica que fará a instalação do sistema. Será analisada a documentação provida pelo PACS em seu site oficial.

- **Customização do software**

Será analisado qual a capacidade do software em ser customizado, ou alterado pela equipe técnica da clínica, para ajustar às suas necessidades.

- **Segurança**

Será realizado um levantamento, para saber como funcionam as permissões de acesso de cada sistema, e se há um controle de usuários e permissões, para que haja segurança sobre as ações de cada usuário que possui acesso ao sistema.

- **Pós-processamento de imagens**

Em alguns casos pode ser necessário realizar algumas ações nas imagens para facilitar o diagnóstico, será identificado em cada software a capacidade em ajustar o contraste, luminosidade, coloração, filtros, rotação, espelhamento e zoom nas imagens adquiridas.

Nesta seção são detalhados os critérios de avaliação utilizados neste trabalho. Todos os itens foram listados anteriormente e explicados de forma sucinta, nos próximos parágrafos, são analisados e descritos mais profundamente, juntamente com as métricas qualitativas e quantitativas a serem consideradas. Os critérios definidos neste trabalho tem como objetivo estabelecer algumas métricas que possam ser utilizadas por profissionais da área da saúde na escolha do melhor sistema PACS para instalar em seu ambiente digital.

Segundo Prodanov (2013), o método quantitativo é empregado, principalmente, quando buscam a relação causa-efeito entre os fenômenos, e também pela facilidade de poder descrever a complexidade de determinada hipótese ou de um problema. Já a abordagem qualitativa difere da abordagem quantitativa pelo fato de não utilizar dados estatísticos como o centro do processo de análise de um problema, não tendo, portanto, a prioridade de numerar ou medir unidades. Os dados coletados nessas pesquisas são descritivos, retratando o maior número possível de elementos existentes na realidade estudada.

Como visto anteriormente, a qualidade em uso de um produto de software é a visão da qualidade percebida sob a perspectiva do usuário. Desta forma, serão analisados alguns critérios que não são medidos quantitativamente apenas, será utilizada uma análise qualitativa em alguns destes, pois são características que dependem da percepção de uso do software pelo usuário.

A metodologia de avaliação dos softwares deste trabalho requer três etapas. A seguir serão apresentadas cada uma das etapas deste trabalho:

- Preparação dos ambientes: instalação e configuração das máquinas virtuais, instalação dos PACS, a fim de entender os requisitos necessários para instalação do sistema, e as funcionalidades disponibilizadas por cada software.

- Coleta dos dados: a primeira tarefa efetuada neste ponto, foi a importação de algumas imagens de exames para os sistemas. Neste passo também ocorreu a utilização básica do sistema, simulando o uso básico de um profissional da medicina no sistema. Efetuam-se buscas, visualizações, verificação do funcionamento das atividades disponíveis.

- Análise dos resultados: sendo esta a última etapa da coleta de dados, onde serão listados e analisados.

4.1.1 Ambiente de Validação

Nesta seção apresenta-se a metodologia utilizada para instalação dos sistemas PACS, e algumas de suas particularidades. A máquina onde serão executados os testes, é um Macbook Pro, late 2013, com 256GB de SSD, 8GB de RAM, e processador Intel Core i7 2GHz, como pode ser conferido na Figura 6.

Figura 6: Especificações da máquina de testes



Fonte: Próprio Autor

Para se obter resultados justos entre os softwares, todos os testes dos sistemas foram realizados em Máquinas virtuais, com sistema operacional Windows XP - devido ao seu baixo requisito de processamento e memória - com 1024MB de RAM reservada, 1 core do processador da máquina onde roda o sistema de virtualização, e 10GB de armazenamento no drive virtual. Cada máquina virtual funcionará como servidor e cliente para os testes. Os PACS Orthanc, Dicoogle, Conquest e dcm4chee são compatíveis com

o sistema operacional Windows, apenas o Horos é desenvolvido para Mac OS, portanto ele será testada na máquina host.

4.1.2 *Workflow* básico de utilização do software

Para fins de comparação, foram executadas tarefas iguais em todos os softwares, para que pudessem ser analisados como se comportam em um cenário controlado. O acesso se deu por meio do software VirtualBox, onde estarão as máquinas virtuais de cada um dos softwares individualmente configuradas.

O roteiro de trabalho utilizado em cada sistema é sempre o mesmo, dividido em seis etapas principais, e dentro destas, separado em itens menores, com base em critérios selecionados com objetivo de responder as perguntas necessárias para exemplificar as características de qualidade da ISO/IEC 9126 (NBR13596), listados abaixo:

1. Instalação

- (a) Consulta à documentação do software para realização do processo de instalação
- (b) Instalação do sistema PACS
- (c) Abrir software
- (d) Verificar configuração inicial (instalação do serviço no windows, liberação de portas)
- (e) Utilização inicial do sistema (verificar se o sistema abre)
- (f) Verificar se há algum controle de usuários do sistema

2. Armazenamento dos arquivos

- (a) Verificar configurações de armazenamento (*filesystem* ou banco de dados)
- (b) Identificar maneiras de *backup*

3. Utilização

- (a) Importação de exames médicos (importar uma quantidade de exames pré-definida)
- (b) Busca de exames (realizar a busca de exames)
- (c) Com os testes realizados, identificar as mensagens de erro e se são claras.
- (d) Avaliar a estabilidade dos softwares durante os testes

4. Comunicação

- (a) Verificar se o PACS utiliza o padrão DICOM para comunicação
- (b) Investigar se há a possibilidade de comunicação via WEB, para consulta de exames (WADO Server)

5. Visualização

- (a) Analisar e avaliar a interface do sistema
- (b) Visualização das imagens (visualizar exames e as opções disponíveis)
- (c) Ver se há opções de manipulação de imagens, zoom, contraste, etc..
- (d) Verificar se há a possibilidade de impressão dos exames através do software

6. Customizações

- (a) Avaliar a possibilidade de customização do software (*plugins*)
- (b) Analisar se há a possibilidade de tradução do software
- (c) Verificar as opções de Diagnóstico Automático

5 Análise dos Softwares

Neste capítulo estão documentados a execução dos testes listados no capítulo anterior, e foram analisados os critérios de avaliação de cada software. A ordem dos subcapítulos segue a ordem do roteiro descrito no capítulo 4.1.1.

Dois PACS não são analisados nas seções subsequentes pois apresentaram dificuldades de instalação, o dcm4chee e o Conquest, e estas dificuldades são abordadas no Capítulo 5.8.

No Quadro 5, ao final do capítulo, pode ser encontrada uma síntese das características analisadas neste trabalho. É possível ter uma visão geral sobre as análises realizadas neste capítulo.

5.1 Documentação

É indispensável que as empresas gerenciem o conhecimento utilizado em seus produtos e serviços de forma adequada para garantir aprendizado, crescimento e preservação do capital intelectual. (DIAS and DA COSTA, 2017)

E não apenas para a empresa desenvolvedora do software, a documentação também auxilia o usuário final a aprender mais rapidamente o funcionamento do produto, diminuindo a curva de aprendizado necessária para utilização do mesmo.

5.1.1 Horos

O Horos é um visualizador de imagens médicas *open-source*, e pode funcionar como um servidor DICOM também. A página oficial do projeto não é muito estruturada de maneira a facilitar o acesso à informação. Não existe uma área específica do site voltada às funcionalidades do software, embora hajam alguns tutoriais mais simples, exemplos de como importar imagens, configurar o acesso à servidores na nuvem, entre outros.

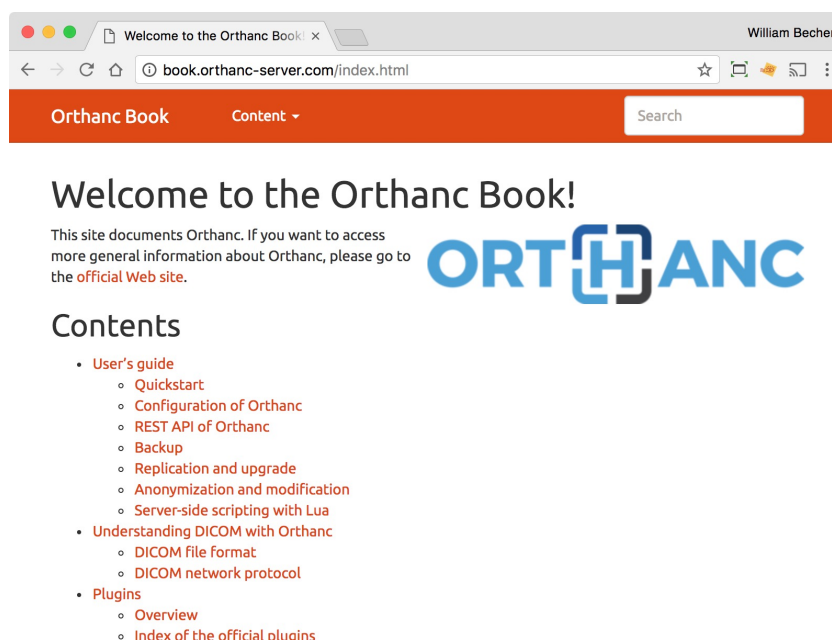
Por ter acesso grátis, o Horos foca a sua maneira de ganhar dinheiro em cursos e treinamentos, e há uma falta de documentação gratuita disponível no seu site. Mas em contrapartida, por ser baseado em uma versão do Osirix, que é pago, a documentação do PACS pago pode ser usada sem problemas para o Horos. Em seu site, o Horos disponibiliza um formulário para solicitação de suporte técnico pago, e cobra 79 dólares por 15 minutos de suporte.

5.1.2 Documentação do Orthanc

A documentação do Orthanc é bem completa. O site possui um tutorial sucinto para realizar a instalação do sistema. Nele, são explicados passo a passo cada tarefa que precisa ser realizada para que o sistema funcione. Para a realização deste estudo, foi seguido esse roteiro de instalação conforme o site.

Após terminar a instalação, que será detalhada no Capítulo 5.2.2, o instalador redireciona o usuário para o site Orthanc Books, como podemos ver na figura 7. Neste site a documentação é bem mais completa, onde pode-se conferir desde uma visão geral sobre o padrão DICOM, bem como algumas configurações mais complexas do sistema.

Figura 7: Screenshot do site Orthanc Books



Fonte: Próprio Autor

Sumário das informações encontradas no site Orthanc Books:

- *User's guide*
 - *Quickstart*
 - *Configuration of Orthanc*
 - *REST API of Orthanc*
 - *Backup*
 - *Replication and upgrade*
 - *Anonymization and modification*
 - *Server-side scripting with Lua*

- *Understanding DICOM with Orthanc*
 - *DICOM file format*
 - *DICOM network protocol*
- *Plugins*
 - *Overview*
 - *Index of the official plugins*
 - *Index of the contributed plugins*
- *Developer's guide*
 - *Creating new plugins*
 - *Stone of Orthanc*
 - *Database versioning*
 - *Implementation notes*
 - *Coding style*
- *Frequently asked questions*
 - *General questions*
 - *Features supported by Orthanc*
 - *Specific FAQs*
- *Contributing to Orthanc*

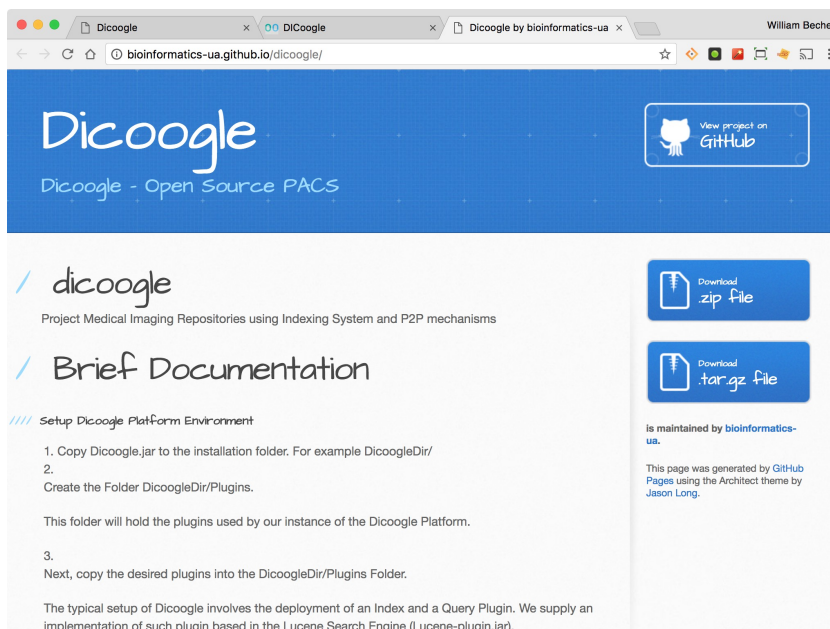
5.1.3 Documentação do Dicoogle

Desenvolvido pelo grupo de bioinformática da Universidade de Aveiro, a documentação do Dicoogle é bem escassa. Quando faz-se a instalação, o próprio instalador abre um arquivo *readme.txt* para o usuário com algumas informações básicas, de diretórios e arquivos de configuração. Como é visto na Figura 8, no site <http://bioinformatics-ua.github.io/dicoogle/> pode ser acessada uma versão *online* deste documento, onde tem-se mais informações sobre o processo de instalação. Nesse site as instruções são bem superficiais, apenas alguns *links* apontando para a página do *github*, onde pode-se observar que não há muitos *commits* no projeto.

Não há uma área específica do site do Dicoogle voltada para a documentação, esse é um ponto fraco deste PACS. Existem alguns artigos publicados e referenciados no

site, mas é difícil para o novo desenvolvedor aprender um pouco mais sobre a arquitetura do software, sem um ponto de partida.

Figura 8: Página da documentação do Dicoogle



Fonte: Próprio Autor

5.2 Instalação e configuração dos softwares

Nesta seção do trabalho serão abordados os processos de instalação e configuração dos softwares para funcionamento. Serão avaliados os processos de instalação, a configuração necessária para o funcionamento inicial, e o controle de acessos através de um controle de usuários.

5.2.1 Horos

A instalação do Horos é feita através de um instalador próprio, que pode ser obtido no site oficial, em <https://www.horosproject.org>, mediante um cadastro simples. Após o término do cadastro, e *download* do software, e seguindo o padrão de instalação de softwares para Mac OS, basta arrastar o programa para a pasta *Applications*, e a instalação está completa.

Não é necessário realizar nenhuma operação posterior à instalação para começar a usar o Horos, basta abrir o aplicativo. Já é disponibilizado um servidor web para acesso aos exames, e um *listener* para a comunicação via padrão DICOM com o Horos.

Uma das vantagens do Horos é já vir disponível a versão 64bits no projeto open-source, enquanto no Osirix, a versão grátis funciona apenas em 32bits, e a 64bits é

disponibilizada somente na versão paga.

5.2.2 Instalação do Orthanc

A instalação do Orthanc foi bem simples, ocorreu normalmente sem apresentar nenhum erro. O instalador pode ser baixado em www.orthanc-server.com/download.php, para diversas plataformas. Existem duas versões do software, uma que instala o servidor como um serviço do Windows, que iniciará junto com a máquina, e outra que é a versão linha de comando, que precisa ser iniciada cada vez que o usuário desejar utilizar o software. A escolhida para este trabalho, foi a versão 17.7.0 para Windows, instalada como serviço. Ao rodar o instalador, aceitar os termos e avançar com a instalação, será criado um serviço no Windows chamado Orthanc, que iniciará com o sistema sempre ao iniciar a máquina.

O acesso ao sistema se dá totalmente via *browser*, via interface web, que se utiliza de *REST API* para o funcionamento e a comunicação entre sistemas.

O restante da configuração pode ser acessada em um arquivo JSON, que fica localizado em *c:\Arquivos de programas\Orthanc Server\Configuration\orthanc.json*. Este arquivo é subdividido em seções específicas, sobre a configuração geral do software, sobre o servidor http, sobre o servidor DICOM, opções de segurança, topologia de rede e outras opções avançadas, como pode ser visto no Quadro 2.

Quanto a existência de um processo de *backup* automatizado, esta não é uma opção do Orthanc, porém é uma tarefa que pode ser feita de maneira simples, caso o armazenamento seja feito no *filesystem*, visto que basta copiar a pasta principal de armazenamento, juntamente com o banco de dados SQLite que serve de índice.

Conforme pode ser visto no arquivo de configuração, o Orthanc é uma opção bastante customizável para ser instalado em ambientes que necessitem de um servidor de imagens *open source* bastante customizável, e extensível. O ponto negativo fica na conta da segurança, já que os usuários e senhas dos usuários fica armazenado em um arquivo texto, o que pode ser considerada uma grande falha de segurança.

5.2.3 Dicoogle

O Dicoogle é um PACS multiplataforma, e não possui instalação. Basta descompactar o arquivo baixado diretamente do site www.dicoogle.com em uma pasta, e rodar o software. Essa forma de instalação é simples, e não requer configurações extras para o software funcionar.

No arquivo *config.xml* são encontradas as configurações do sistema, como portas de acesso, caminhos de leitura e armazenamento, configurações de indexação, nome do

Quadro 2: Seções do arquivo de configuração do Orthanc

Seção	Configuração
Configurações Gerais	Configurações gerais do software. Diretório para armazenamento de arquivos, diretório para armazenamento do banco de dados SQLite, compressão de dados, tamanho máximo de armazenamento em disco, quantidade de pacientes permitidos e lista de plugins instalados.
Configurações do Servidor HTTP	Ativa ou desativa o servidor HTTP, porta onde será instalado o servidor e a API REST, ativação de erros na tela e compressão HTTP.
Configurações do Servidor DICOM	<i>Encode</i> padrão dos arquivos DICOM, sintaxes de transmissão aceitas pelo Orthanc C-STORE, tempo de <i>timeout</i> .
Configurações de Segurança HTTP	Permissão de acesso remoto ao servidor, conexão SSL, configurações de autenticação, lista de usuários e senhas.
Configurações de Topologia de Rede	Configuração de outros <i>peers</i> na rede.
Configurações avançadas	Configurações de detalhes sobre o funcionamento do servidor, armazenamento de MD5 dos arquivos no banco de dados, limite máximo de resultados de busca, limite de <i>jobs</i> simultâneos do servidor, opção de armazenar ou não os arquivos DICOM, entre outras.

Fonte: Próprio Autor

serviço, entre outras. Esse arquivo é um XML estruturado, mas não apresenta nenhum comentário ou documentação acerca das funções contidas nele, o que pode tornar difícil a customização do servidor DICOM logo de início.

O Dicoogle possui um arquivo XML de configuração separado para o cadastro de usuários, a estrutura deste arquivo é fácil de ser entendida: dentro da tag *Users*, podem existir quantos usuários forem necessários pelo sistema. Cada tag *user* contém parâmetros com as informações de nome de usuário, senha e um valor booleano para indicar se o usuário possui direitos de administrador ou não. O nome de usuário e os direitos são salvos como *plain-text*, já a senha é criptografada com um *hash*, dificultando o vazamento de senhas. Em comparação com o cadastro de usuários do Orthanc, esta é uma abordagem mais segura.

5.3 Configurações de Armazenamento

Nesta seção do trabalho será descrito o funcionamento dos PACS analisados com relação ao armazenamento e indexação das informações dos arquivos DICOM processados

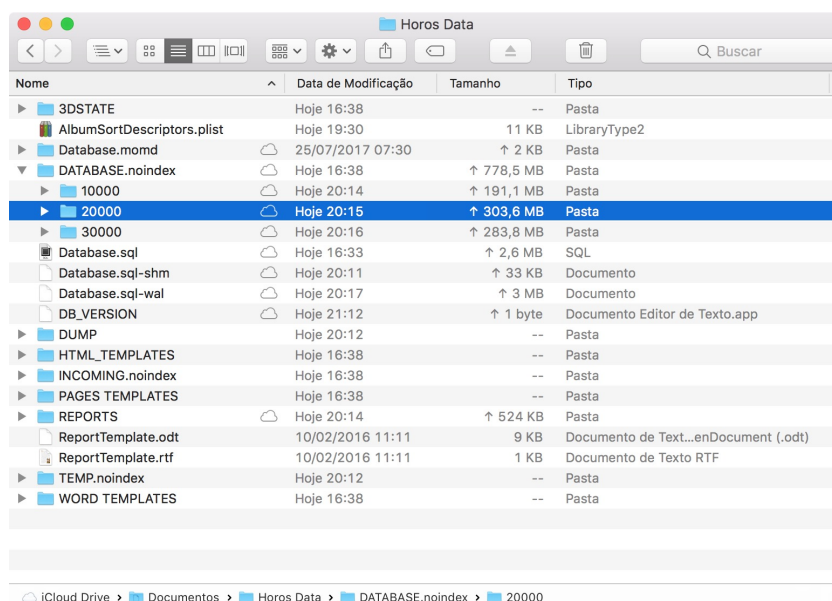
pelos servidores.

5.3.1 Horos

Ao instalar o Horos, ele disponibiliza um diretório nos arquivos pessoais do usuário, dentro da pasta Documentos, onde será salva toda a estrutura de arquivos necessários para o funcionamento do software, incluindo os arquivos DICOM. Ao importar arquivos de imagens médicas no Horos, o usuário tem a opção de movê-las para a pasta padrão de armazenamento, ou deixá-las na pasta original.

Numa instância de um servidor Horos, pode-se trabalhar com mais de um diretório de banco de dados, de maneira a organizar melhor o fluxo de trabalho, e até mesmo para manter um número não muito grande de estudos em cada diretório para que não haja nenhuma perda de velocidade de processamento ao utilizar o software em diretórios com vários exames com muitas imagens, que desaceleram o controle do sistema operacional aos arquivos.

Figura 9: Captura de tela com as pastas de arquivos do Horos



Fonte: Próprio Autor

Os arquivos físicos, são organizados em pastas numeradas, como pode ser visto na Figura 9, com no máximo dez mil arquivos, e o índice é mantido em um arquivo binário chamado *Database.sql*, na pasta raiz. Embora o Horos seja indicado em seu site oficial como um visualizador DICOM, ele pode muito bem suportar a carga de um pequeno servidor de armazenamento de imagens médicas.

O processo de *backup* do Horos funciona melhor se for feito utilizando o próprio sistema, já que possui uma função específica para a exportação dos arquivos dele para

uma pasta específica, e os arquivos DICOM podem estar em diversas pastas linkadas, caso o usuário não os importe fisicamente para o diretório de instalação. O processo de *backup* nativo do Horos pode tornar-se demorado se executado em bases com muitos exames, pois o usuário precisa selecionar todos os exames e clicar no botão de exportação.

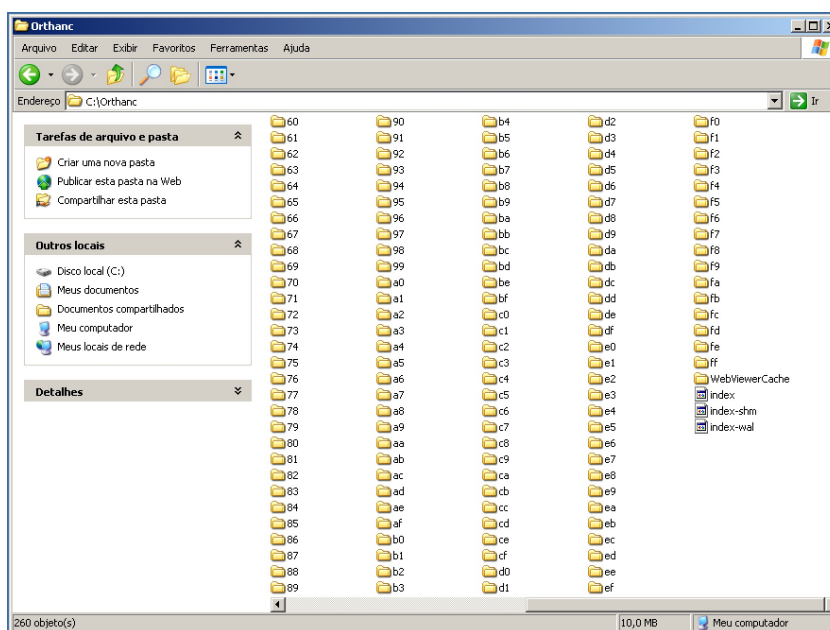
5.3.2 Orthanc

Por padrão, e como é o caso deste trabalho, se não houver nenhum banco de dados disponível, o Orthanc armazena todos os arquivos DICOM recebidos em um diretório, chamado OrthancStorage no sistema de arquivos. Para facilitar a organização, é criado, juntamente com cada arquivo DICOM, um JSON que contém todas as *tags* relevantes do arquivo, de maneira a facilitar o processamento e busca dos dados.

Para economizar espaço, o PACS pode ser configurado para que automaticamente comprima as imagens assim que as receber. Para reduzir a quantidade de arquivos em um diretório único (algo que alguns sistemas operacionais têm dificuldade em processar), uma hierarquia de 3 níveis de diretórios é criada para armazenar os anexos: os primeiros dois caracteres hexadecimais do UUID (*Universally Unique Identifier*), do arquivo são o primeiro nível, e os próximos dois caracteres formam o segundo nível. Na Figura 10 se vê a estrutura de arquivos do servidor.

Dentro do diretório de armazenamento, é mantido um banco de dados SQLite, chamado de índice, que indexa todos os arquivos utilizados pelo sistema. Esse banco de dados grava diversas informações sobre os arquivos armazenados pelo Orthanc, como o método de compressão da imagem, os *hashes* MD5 de antes e depois da compressão para detectar qualquer arquivo que possa estar corrompido.

Figura 10: Estrutura de pastas do Orthanc Server



Fonte: Próprio Autor

O funcionamento padrão do Orthanc é esse, armazenamento no *filesystem*, e indexação dos arquivos em uma base de dados SQLite. Mas o desenvolvedor já disponibiliza plugins para uma substituição desses métodos de armazenamento mais simples, por uma base PostgreSQL, tanto para a indexação dos arquivos, quanto o *storage* dos arquivos DICOM. A alteração é simples e requer além da instalação do banco de dados PostgreSQL, a alteração de algumas linhas no arquivo de configuração do servidor e o reinício do serviço.

O processo de *backup* deste servidor é simples, pode ser feito apenas copiando a pasta *OrthancStorage* para um local seguro, a restauração pode ser efetuada copiando os arquivos de volta ao local original. O *backup* dos arquivos DICOM podem ser armazenados na nuvem, para liberar espaço em disco, porém o desenvolvedor recomenda que o arquivo de índice SQLite seja mantido no disco rígido da máquina, visto que é acessado muitas vezes, e o *delay* em alguma atualização do arquivo no processo de envio/retorno da nuvem pode corromper o arquivo de índice.

Estas características citadas acima, tornam o Orthanc uma opção simples e robusta para ser adotada em ambientes que necessitem de um PACS com custo baixo, e manutenção fácil.

5.3.3 Dicoogle

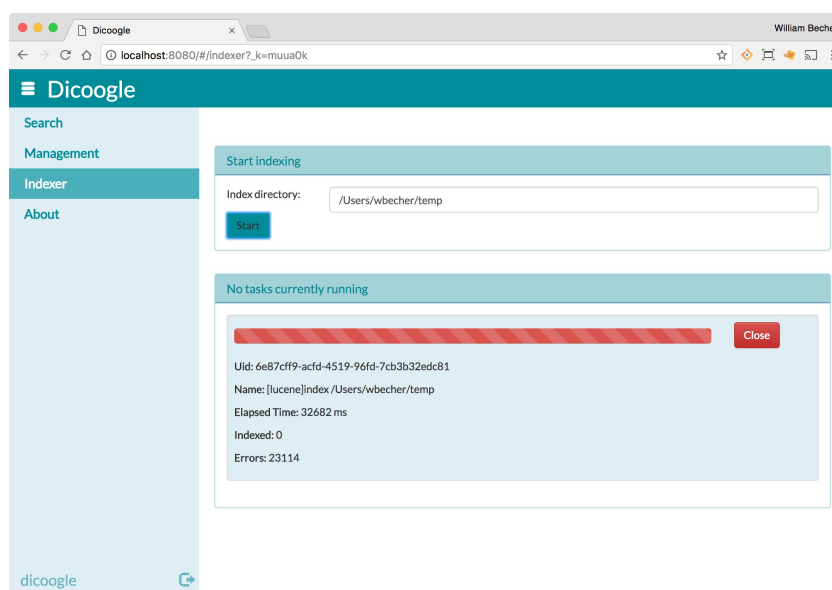
Atualmente, o Dicoogle dispõe de uma solução baseada na biblioteca Apache Lucene. No entanto, esta tem demonstrado alguns problemas de desempenho em ambientes

em que se tem necessidade de manusear e pesquisar sobre uma grande quantidade de dados, muito particularmente em cenários de análise de dados. (ALVES, 2016)

De acordo com Alves (2016), os *plugins* de indexação organizam os dados em um formato que permite o acesso rápido, incluindo os processos de extração e armazenamento de dados. Os modelos de dados utilizados são definidos pelo desenvolvedor dos *plugins*. O acesso aos dados ocorre através de uma API web, acessível através da interface do Dicoogle. O Dicoogle não importa os dados para uma pasta específica, ao invés disso, ele utiliza um sistema de índice, que é atualizado cada vez que uma tarefa de indexação é chamada. Essa tarefa de indexação recebe o caminho para o diretório onde os arquivos DICOM estão armazenados, e então começa a catalogar as imagens, e ao final do processo mostra um relatório na tela com o número de imagens indexadas, erros e tempo total de execução, conforme pode ser visto na Figura 11.

Nenhum arquivo é copiado para o diretório do Dicoogle, todos ficam armazenados na pasta indicada no processo de indexação. No diretório de instalação o servidor mantém uma pasta com um índice, utilizado para acelerar as pesquisas, mostrando resultados sem precisar pesquisar no *filesystem* a cada nova busca.

Figura 11: Tela de progresso do indexador do Dicoogle



Fonte: Próprio Autor

Não há um processo de *backup* automatizado no Dicoogle, portanto, fica a cargo do administrador de sistemas a realização de cópias de segurança. Este processo pode se tornar custoso, a medida que não há um diretório centralizado para armazenamento de arquivos, de maneira padrão no software. Caso o administrador deseje, pode trabalhar com um diretório de armazenamento padrão como convenção, para facilitar este trabalho.

5.4 Roteiro de Utilização

Nesta seção do trabalho, será descrito o roteiro de trabalho dos softwares, e a forma de utilização de cada um deles. Para realização dos testes, foram importados nos PACS, 5,66GB de arquivos DICOM. Totalizando 23.333 arquivos .dcm, de 10 pacientes diferentes, cada um com mais de um exame. A importação dos exames foi feita utilizando a interface de *upload* de imagens do próprio software analisado. As imagens médicas utilizadas neste trabalho foram conseguidas no repositório da *National Biomedical Imaging Archive*, que pode ser acessado através deste link <https://imaging.nci.nih.gov/>.

Em cada software serão importadas a mesma quantidade de exames, que pode ser vista no Quadro 3:

Quadro 3: Distribuição de exames e quantidade de arquivos

ID do Paciente	Qtde de Exames	Qtde de Séries	Qtde de Imagens
0522c0001	4	13	2060
0522c0002	5	16	3393
0522c0003	5	16	3439
0522c0009	5	16	4245
0522c0012	3	11	662
0522c0013	5	16	3447
0522c0014	7	14	1492
0522c0015	4	31	2593
0522c0017	4	25	2273
0522c0027	3	12	1914

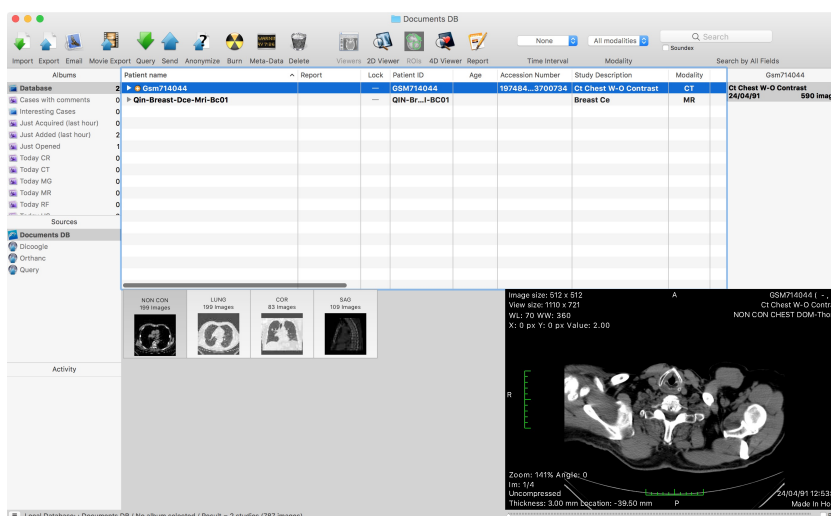
Fonte: Próprio Autor

5.4.1 Horos

Ao abrir o Horos, já se vê a tela inicial da aplicação, mostrada na Figura 12, com a lista de pacientes no centro da tela, a barra de funções no topo, e uma seção de pré-visualização no canto inferior direito.

A primeira tarefa que precisa ser realizada é a importação de exames. Para executar esta tarefa, basta utilizar o botão *import* e selecionar a pasta onde se encontram os arquivos DICOM. Neste momento o software pergunta se o usuário deseja importar os arquivos para a pasta de banco de dados do Horos, ou importar apenas os *links* para os arquivos em sua pasta original. Esta segunda opção economiza espaço em disco, já que não duplica os arquivos no armazenamento. Ao invés do botão *import*, pode-se arrastar imagens para a tela do aplicativo, que ainda assim a função de importar é ativada, e as opções de importação de arquivos ou *links* aparecem na tela.

Figura 12: Tela Inicial do Horos



Fonte: Próprio Autor

Ao selecionar um exame na lista, a pré-visualização é mostrada automaticamente no canto inferior direito, e caso haja mais de uma imagem no exame, o usuário pode apertar espaço para visualizar uma animação com a sequência de imagens dos exames, o que permite uma rápida identificação do conteúdo do exame. E com 2 cliques na linha do exame, o Horos abre a tela padrão de visualização 2D, onde existem opções mais avançadas, e o exame aparece em tamanho maior, ocupando a maior parte da tela. As opções de visualização disponíveis são abordadas no Capítulo 5.6.1.

5.4.2 Orthanc

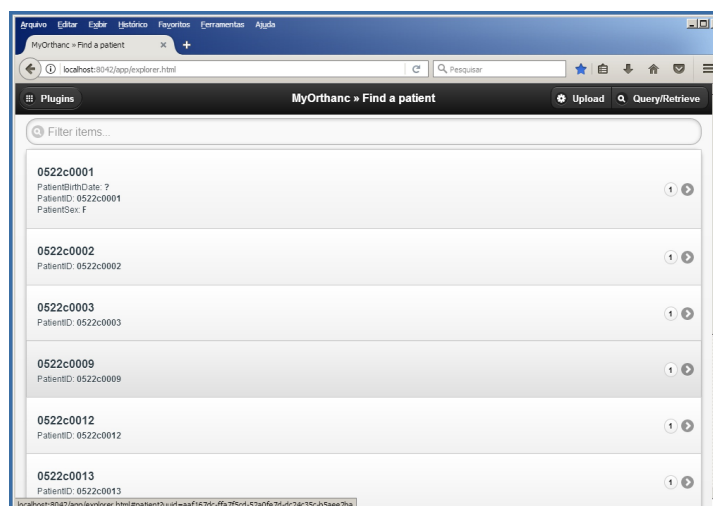
Ao executar o Orthanc, a interface web é o modo padrão de acesso ao banco de dados de imagens. A primeira tarefa executada é a indexação de arquivos, onde serão importadas os arquivos DICOM utilizadas nos testes deste trabalho.

Na opção de *upload* nativa deste PACS, a utilização segue o padrão difundido hoje na web, arrastar os arquivos que deseja adicionar ao banco de dados e clicar no botão para iniciar o *upload*. Neste momento, o servidor começa a importar os arquivos e mostra uma barra de progresso na tela com o progresso atual. Este procedimento é bastante custoso para a máquina virtual, visto que o servidor precisa fazer toda a organização das imagens no sistema de arquivos, e a geração do índice com todas as informações dos exames importados. Em alguns testes de importação o servidor travou ao importar muitas imagens de uma vez, então os arquivos foram importados de cinco em cinco mil, a cada operação de *upload*.

Com os exames já importados, na tela inicial, conforme pode ser visto na Figura 13, é mostrada uma lista de pacientes, pronta para consulta, o botão de *plugins*, o botão

para acesso ao procedimento de *upload*, e o botão de busca. A interface é simples, e limpa, não há dificuldade em entendê-la, já que segue alguns padrões de projeto comuns e na web.

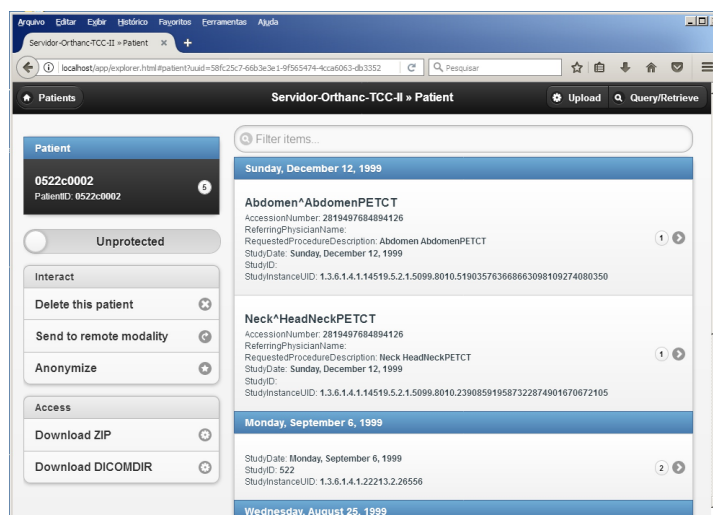
Figura 13: Tela Inicial do Orthanc



Fonte: Próprio Autor

A lista de pacientes toma a maior parte da tela, e mostra diversas informações em cada item. Ao clicar no paciente, o servidor encaminha o usuário para uma lista de exames do paciente escolhido. Esta tela, vista na Figura 14, lista todos os estudos daquele paciente, e ao clicar no estudo, uma tela similar é apresentada, onde o usuário pode selecionar o exame individualmente, e por fim abrir o visualizador web. São disponibilizadas duas versões de visualizadores de imagens, o *Orthanc Web Viewer* e o *Osimis Web Viewer*. Os dois serão abordados de forma mais aprofundada na seção 5.6.2.

Figura 14: Tela da listagem de exames de um paciente no Orthanc



Fonte: Próprio Autor

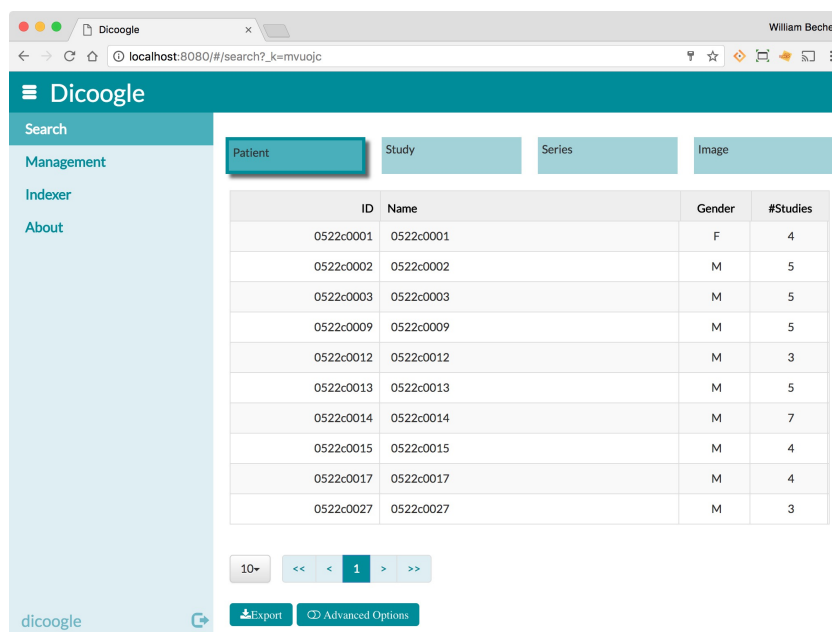
Quando configurado usuário e senha para acesso ao sistema, a utilização se tornou instável. Não foi possível logar no sistema quando ativada a opção de autenticação disposta na seção de segurança do arquivo de configuração descrito na Tabela 2.

5.4.3 Dicoogle

Segundo Alves (2017), após a indexação dos arquivos no Dicoogle, descrito no capítulo 5.3, a tela inicial do Dicoogle é apresentada, com uma interface simples. Quando o botão *search* é clicado, o *plugin* de busca é chamado. Os resultados retornados, são identificados através de suas *tags* DICOM. Os exames são agrupados por Paciente, Estudo, Série (modalidade) e finalmente Instância de exame. Além da lista dos exames é retornado também dados relevantes do paciente. Neste trabalho, todos os dados são anonimizados. A informação coletada pelo *plugin* de busca é convertida em resultados e mostrados de maneira hierárquica, permitindo que o usuário selecione primeiramente o paciente que está buscando, depois selecionar o estudo, depois a série, e finalmente a instância do exame que deseja utilizar.

Na Figura 15, a seguir, é apresentada a tela de resultados de busca do Dicoogle. A navegação não é tão fluida, e por vezes parece muito simplória, sem muitas opções de visualização.

Figura 15: Resultados de busca do Dicoogle - Pacientes



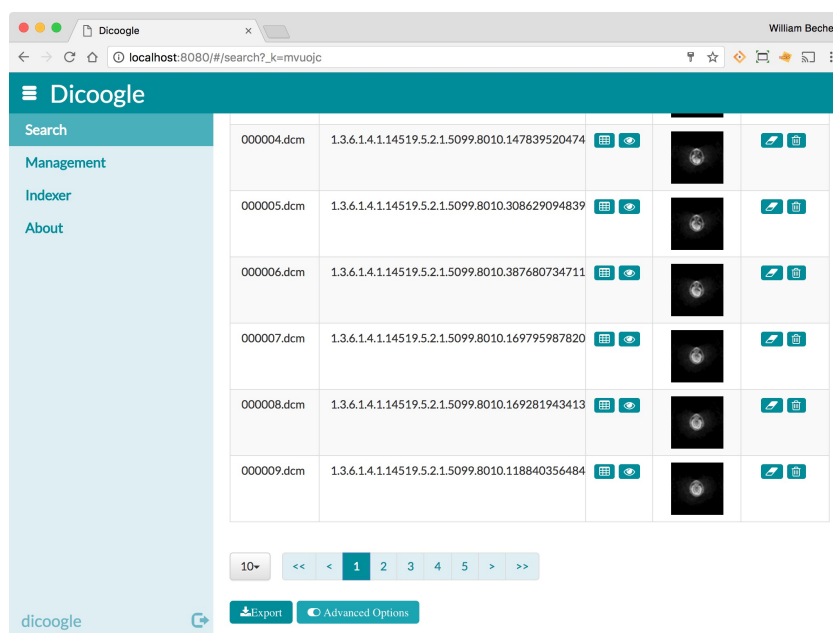
ID	Name	Gender	#Studies
0522c0001	0522c0001	F	4
0522c0002	0522c0002	M	5
0522c0003	0522c0003	M	5
0522c0009	0522c0009	M	5
0522c0012	0522c0012	M	3
0522c0013	0522c0013	M	5
0522c0014	0522c0014	M	7
0522c0015	0522c0015	M	4
0522c0017	0522c0017	M	4
0522c0027	0522c0027	M	3

Fonte: Próprio Autor

Como percebe-se na Figura 16, as opções de visualização de imagens são limitadas, há apenas uma opção de pré-visualização da miniatura, e não há possibilidade de zoom,

sem ser o padrão do navegador. Existem dois botões ao lado esquerdo da miniatura do exame, que servem para mostrar as *tags* DICOM do exame, e o outro é a pré-visualização da imagem, que abre em um *pop-up* na janela do navegador, com a exibição limitada pela área da tela do usuário. Esse é um ponto fraco do Dicoogle, que deixa a desejar no quesito usabilidade e visualização.

Figura 16: Resultados de busca do Dicoogle - Imagens



Fonte: Próprio Autor

5.5 Métodos de Comunicação

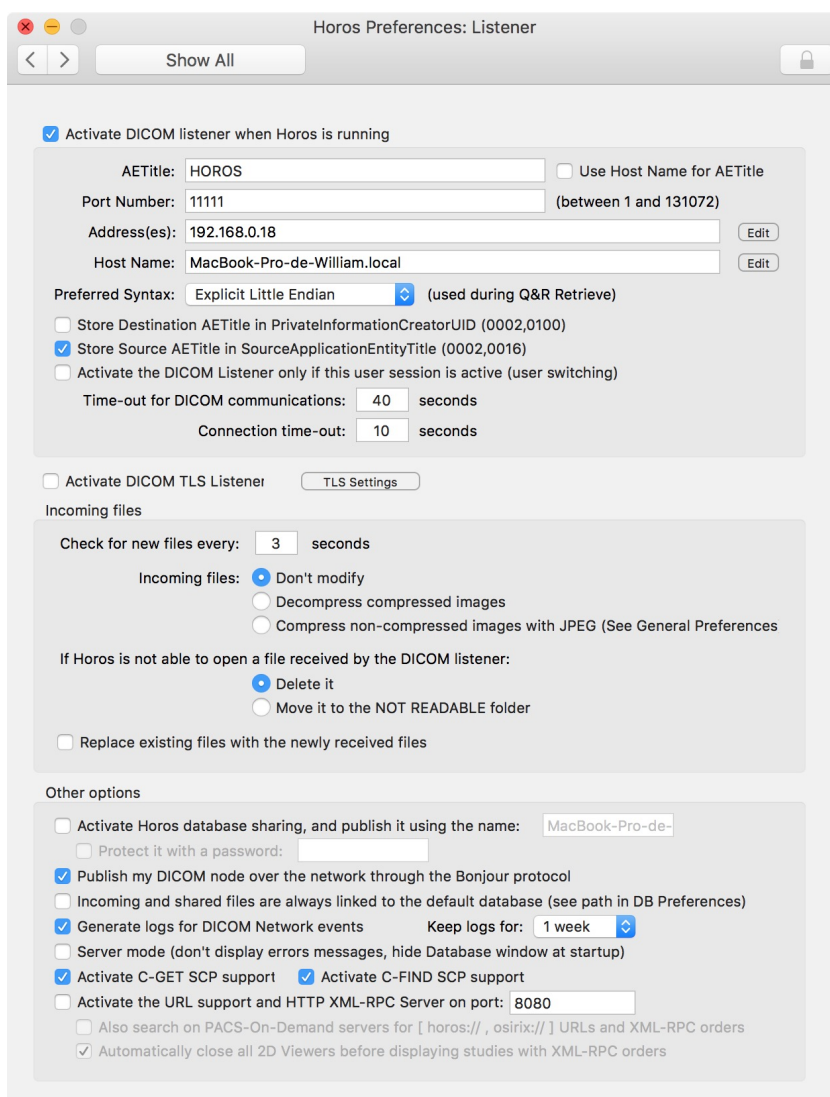
Nesta seção, serão abordados os métodos de comunicação dos sistemas.

5.5.1 Horos

Ao iniciar a utilização do Horos, já é liberado um servidor web, e uma conexão *listener* para a comunicação DICOM. Isso possibilita a utilização do software como um servidor PACS completo. Apesar do site oficial apresentá-lo como um visualizador DICOM, as funcionalidades disponibilizadas pelo Horos permitem esta utilização como servidor.

As configurações são de fácil entendimento, como pode ser notado na Figura 17, e simples de modificar, visto que há interface gráfica, o que facilita realizar as alterações necessárias. Basta configurar a porta de acesso e o endereço IP para o serviço de *listener* DICOM estar funcionando. Com isso, outros nós da rede de computadores poderão utilizar esta instância do Horos como servidor DICOM para receber ou enviar arquivos de exames.

Figura 17: Tela de configuração do *Listener* DICOM do Horos

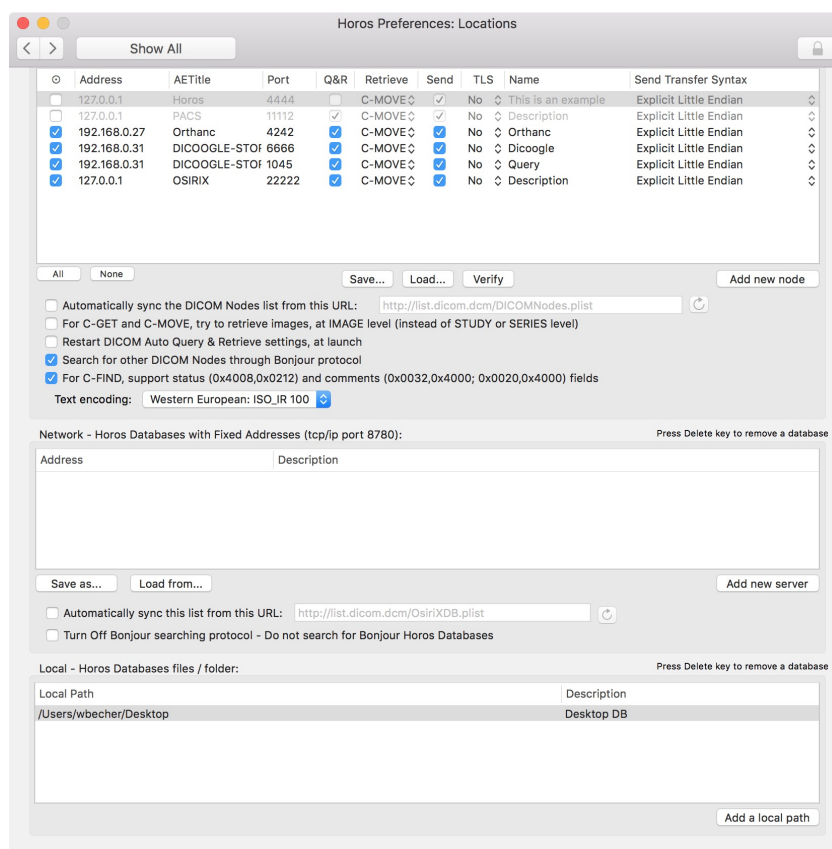


Fonte: Próprio Autor

E ainda nas configurações de comunicação, podem-se adicionar nós na configurações de locais de servidores DICOM, como pode se ver na imagem 18. Ao cadastrar servidores nesta aba da configuração, o Horos será capaz de se comunicar com esses servidores para efetuar busca de exames e envio de arquivos entre os serviços.

Essa gama de ferramentas de rede permitem que o Horos possua protocolos de comunicação que fazem possível a busca, o envio, a solicitação e recebimento de imagens em uma rede PACS, com servidores e máquinas distintas, de maneira simples e sem necessidades de configurações avançadas.

Figura 18: Tela de configuração de locais DICOM do Horos



Fonte: Próprio Autor

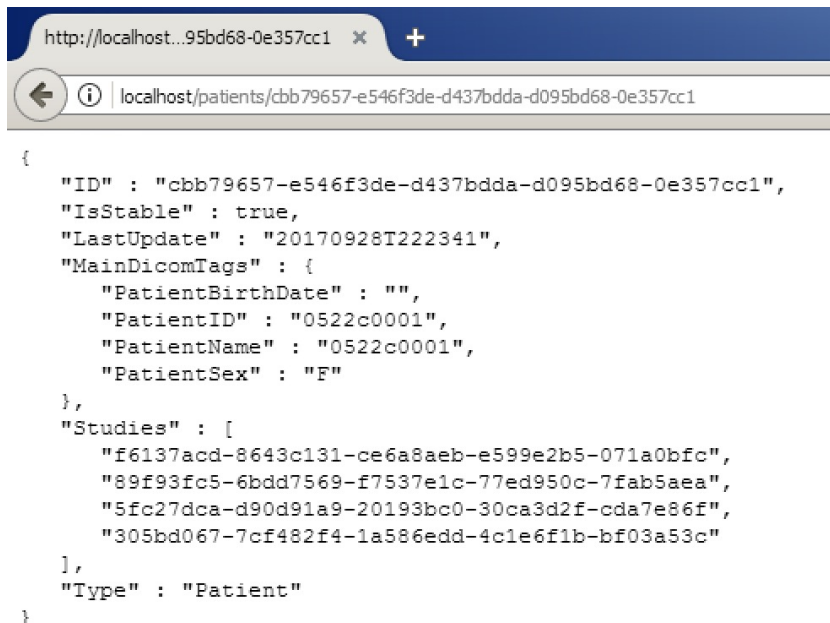
5.5.2 Orthanc

Segundo Orthanc (2017), uma das características mais interessantes deste PACS é a sua *API REST*, que pode ser usada como meio de comunicação entre o servidor DICOM e aplicativos de terceiros, independente da linguagem de programação em que esta aplicação externa está escrita. A API do Orthanc foi escrita para dar acesso total aos recursos do núcleo do servidor. É importante salientar que todas as funcionalidades do Orthanc Explorer (a interface web nativa do Orthanc) utilizam esta API, ou seja, tudo que o Orthanc Explorer faz, também pode ser feito através de *queries* a esta *API REST*. A seguir, na Figura 19, pode ser visto um exemplo de *query*, utilizando o *browser*, para gerar o arquivo JSON de resposta na tela.

Os principais comandos suportados pela interface *REST API* do Orthanc podem ser visualizados *online*, em uma planilha mantida atualizada pela equipe de desenvolvimento, e pode ser acessada no link <http://bit.ly/OrthancRESTAPI>.

Esta funcionalidade é muito útil, e bastante difundida pelo desenvolvedor do Orthanc. Quem está interessado em utilizar este PACS como servidor de sua aplicação tem a possibilidade de estender o funcionamento do sistema e criar sua própria interface

Figura 19: Resposta de uma *query* à API do Orthanc Server



```

{
  "ID" : "cbb79657-e546f3de-d437bdda-d095bd68-0e357cc1",
  "IsStable" : true,
  "LastUpdate" : "20170928T222341",
  "MainDicomTags" : {
    "PatientBirthDate" : "",
    "PatientID" : "0522c0001",
    "PatientName" : "0522c0001",
    "PatientSex" : "F"
  },
  "Studies" : [
    "f6137acd-8643c131-ce6a8aeb-e599e2b5-071a0bfc",
    "89f93fc5-6bdd7569-f7537e1c-77ed950c-7fab5aea",
    "5fc27dca-d90d91a9-20193bc0-30ca3d2f-cda7e86f",
    "305bd067-7cf482f4-1a586edd-4c1e6f1b-bf03a53c"
  ],
  "Type" : "Patient"
}

```

Fonte: Próprio Autor

de acesso, integração com outros sistemas, tornando uma ótima opção para clínicas que desejam uma opção customizável.

5.5.3 Dicoogle

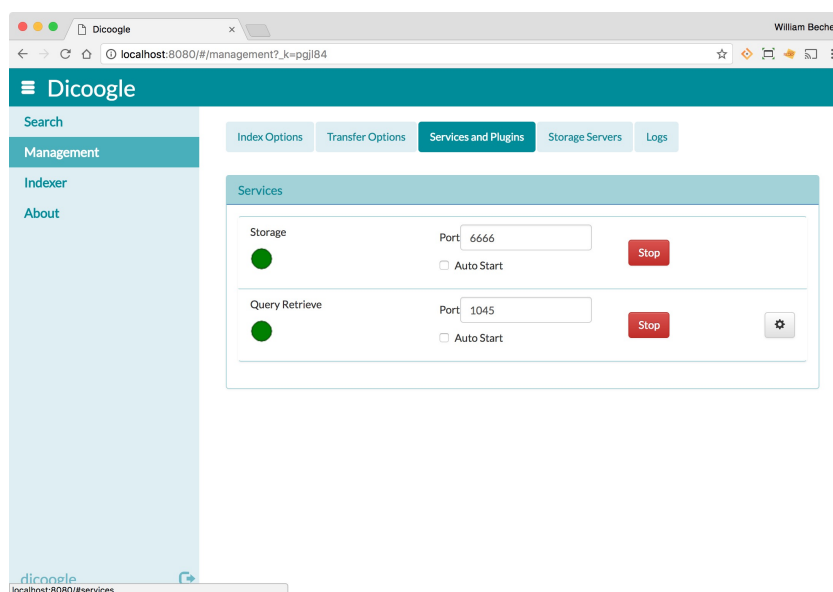
Segundo Dicoogle (2017), o servidor suporta nativamente os padrões de *storage* e *query/retrieve*. Ele também provê um *frontend web* para navegação e uma API para acessar os arquivos DICOM indexados. A interoperabilidade entre o PACS e outras soluções de visualização é disponibilizada através de serviços DICOM, mas outras soluções são suportadas também.

A navegação no *frontend*, e os serviços DICOM foram testados neste trabalho e funcionaram de maneira fácil, bastando ativar os serviços nas configurações, apresentadas na Figura 20. A API web porém não apresenta nenhuma documentação, o que impossibilitou a realização de testes de integração web.

5.6 Visualização de Exames

A visualização dos exames com rapidez e qualidade é algo imprescindível para o bom diagnóstico executado pelo profissional utilizando o sistema de arquivos. Nesta seção será descrito o processo de visualização de exames, informações disponíveis nos softwares, e opções de edição à disposição do usuário final.

Figura 20: Tela de configuração dos serviços DICOM do Dicoogle



Fonte: Próprio Autor

5.6.1 Horos

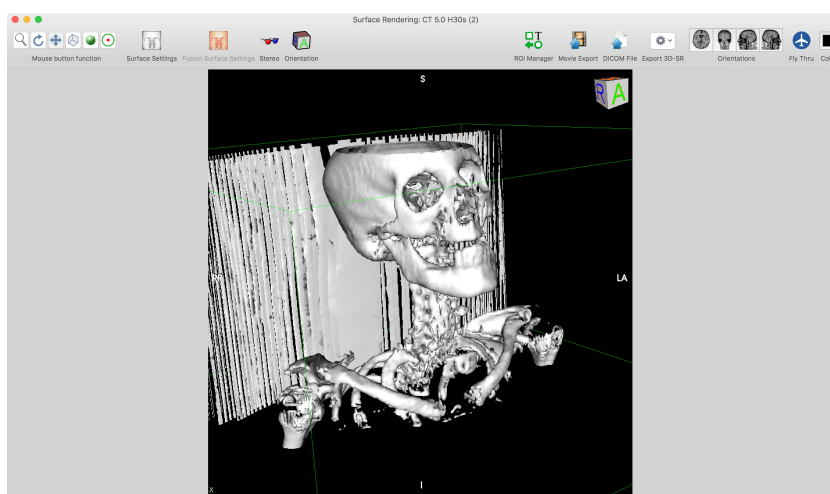
A visualização de exames com o Horos é um dos pontos mais fortes do software. Construído em uma plataforma da Apple, ele se utiliza de um ambiente de desenvolvimento completo, que facilita o rápido desenvolvimento de aplicações gráficas. Como Rosset (2004) afirma em seu artigo, o OsiriX, projeto que deu base ao Horos, foi desenvolvido em menos de 6 meses graças a esse ambiente de desenvolvimento facilitado para o tratamento de imagens, e construção de interfaces.

Quanto à visualização de exames, o Horos também ganha muito em performance, por ser um software específico para MacOS, ele acaba se beneficiando da integração que os sistemas da Apple possuem com o hardware em que são instalados. E ao se utilizarem de bibliotecas gráficas conhecidas, apenas reforçam esta vantagem. O Horos se utiliza da biblioteca OpenGL (2017), que é amplamente difundida hoje em dia e compatível com placas da NVIDIA e ATI, permitindo que o usuário consiga utilizar a aceleração gráfica 3D das placas de vídeo via hardware. Ao adotar o uso da OpenGL, o Horos consegue mostrar grandes conjuntos de 1000 imagens em poucos segundos. (ROSSET et al., 2004)

O Horos também utiliza um *kit* de funcionalidades chamado VTK (2017), que é uma biblioteca para auxiliar a visualização de imagens. Essa biblioteca é *open-source*, distribuída gratuitamente, e é utilizada por milhares de pesquisadores e desenvolvedores ao redor do mundo, justamente por permitir o acesso facilitado à diversas funções de manipulação de imagens, em sistemas multi-plataforma. Por ser desenvolvida sob a licença *open-source* também, o VTK está sempre evoluindo, com a adição de novas funcionalidades, pela comunidade. (VTK, 2017) O VTK oferece funções muito úteis

para o Horos, e por ser construída sobre o OpenGL, assegura que todas estas funções se utilizem da aceleração de hardware dos equipamentos. A manipulação e visualização de *sets* de imagens com dados 3D é uma de suas funções, e disponibiliza isto com alguns algoritmos: Reconstrução Multiplanar (MPR), projeção de intensidade máxima (MIP), renderização de volume com transparência, renderização de superfície com texturização 2D ou 3D. Pode se notar na Figura 21 a tela de visualização 3D com renderização de superfície.

Figura 21: Visualização 3D com renderização de superfície do Horos



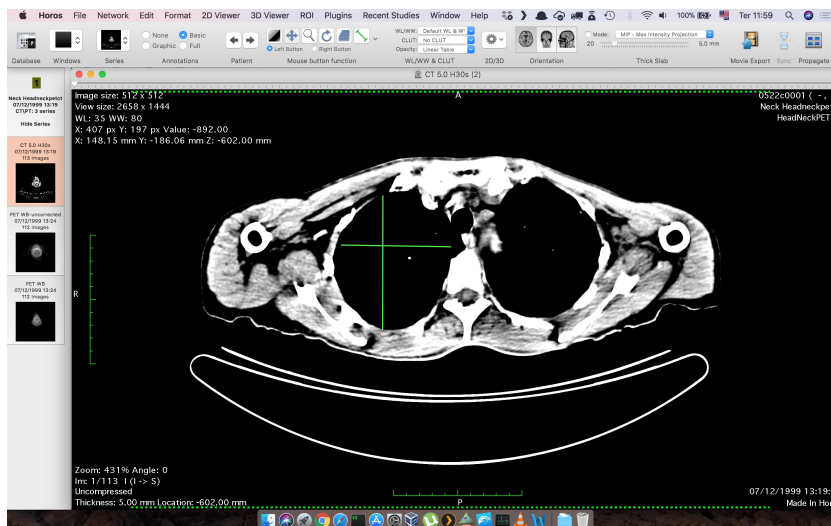
Fonte: Próprio Autor

Um grande trunfo do Horos é a capacidade de exportação de imagens em formatos multimídia que são considerados padrões. Isso acaba aumentando a possibilidade de integração do software em ambientes distintos. No Horos, é utilizada a biblioteca Quicktime (2017), que permite a exportação de exames em qualquer formato, tais como: TIFF, Photoshop, JPEG, JPEG2000, BMP e ainda formatos de vídeo, como AVI, MPEG e MPEG4. E com a biblioteca Quicktime, os desenvolvedores conseguiram ainda habilitar a importação de imagens não-DICOM no Horos, permitindo que o usuário se beneficie das ferramentas de processamento e visualização do software para estas imagens. (ROSSET et al., 2004)

Na Figura 22 pode ser visto a tela de visualização de exames em 2D, com as opções dispostas na barra de botões na parte superior da tela. Das opções disponíveis, destacam-se a possibilidade de visualização de exames lado a lado, opções de zoom e posicionamento, opções de visualização de cores, contraste e opacidade dos exames, exportação de vídeos, e opções de plotagem dos exames em modelos 3D, com a utilização da biblioteca VTK. Na visualização dos exames, existem ainda opções que são ativadas quando algumas teclas são pressionadas, ao segurar o *Ctrl*, são apresentadas as informações adicionais do exame, tais como data e hora da realização, e com o *Shift*,

habilita-se uma lupa no cursor do mouse que pode ser utilizado para dar destaque em pontos específicos da imagem.

Figura 22: Visualização 2D do Horos

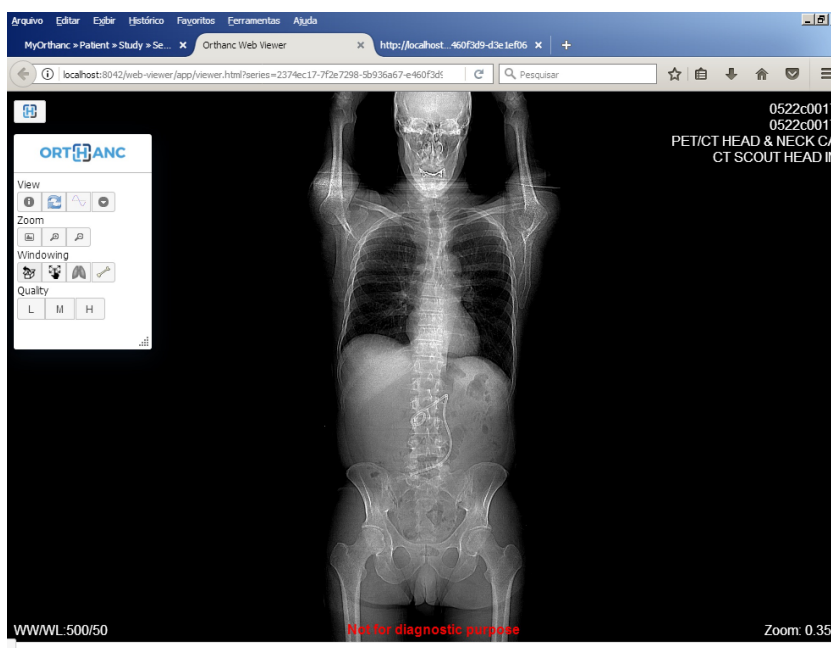


Fonte: Próprio Autor

5.6.2 Orthanc

O Orthanc disponibiliza dois visualizadores para seus exames. Nos testes realizados, apenas o *Orthanc Web Viewer* foi possível ser utilizado. A outra opção, o *Osimis Web Viewer* não funcionou. Nas ferramentas de inspeção do navegador, foi possível detectar que estava faltando um arquivo de configuração da interface web, portanto apenas o *Orthanc Web Viewer* será analisado nesta seção. Na Figura 23 pode ser vista a interface do visualizador padrão de imagens.

Figura 23: Tela de visualização de exames do Orthanc Web Viewer



Fonte: Próprio Autor

As opções do Web Viewer do Orthanc são básicas se comparadas com as do Horos, mas são úteis, e podem ser visualizadas no Quadro 4. As opções mais úteis são as de *zoom*, e ajuste de contraste de imagens. Mas falta uma opção para a navegação na própria imagem quando o *zoom* está aproximado. É possível aumentar o *zoom* apenas na área central da imagem, o que acaba sendo pouco útil para analisar um exame inteiro.

Quadro 4: Opções disponíveis no visualizador web do Orthanc

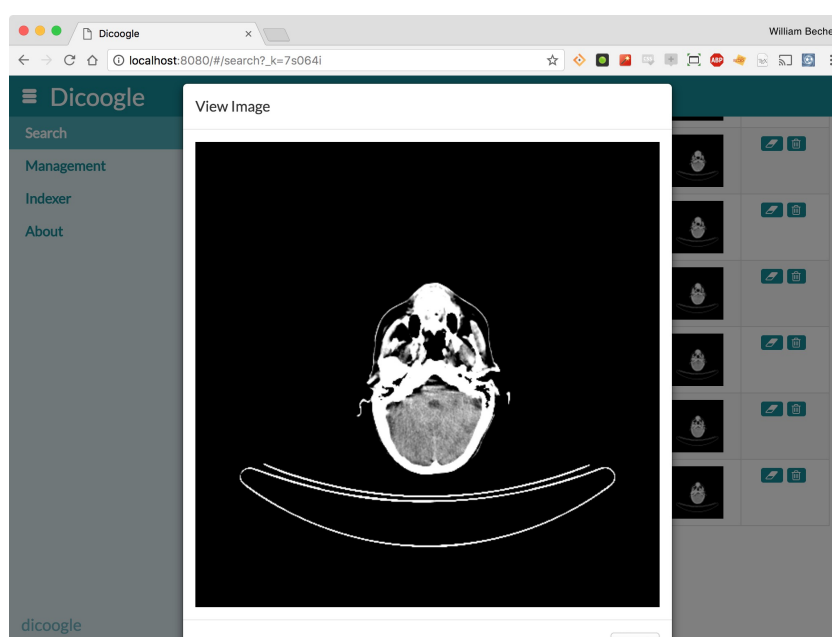
Seção	Opções	Função
View	Series Information	Ver/Ocultar informações sobre o exame
	Reverse Contrast	Inverte o contraste da imagem
	Interpolation	Muda o modo de renderização da imagem
	Download DICOM	Baixar arquivo .dcm
Zoom	Adjust	Retorna ao zoom normal
	Zoom In	Aproxima o Zoom
	Zoom Out	Afasta o Zoom
Windowing	Default Windowing	Ajustes de contraste
	Stretch Contrast	Ajustes de Contraste
	Lung	Modo de contraste para visualização de pulmões
	Bone	Modo de contraste para visualização de ossos
Quality	Low	Renderiza imagens menores, com qualidade baixa
	Medium	Renderiza imagens de qualidade média
	High	Renderiza imagens de alta qualidade

Fonte: Próprio Autor

5.6.3 Dicoogle

O Dicoogle não possui uma interface de visualização tão completa quanto as outras soluções analisadas.. A interface padrão é simplória, conforme pode ser notado na Figura 24, e não possui opções avançadas de tratamento de imagem, ou qualquer tipo de ajustes. Ao clicar na opção de visualização, a imagem é mostrada em um modal *popup*. A única possibilidade de melhoria na visualização é aplicar o zoom nativo do navegador, o que pode ocasionar a perda de qualidade na imagem.

Figura 24: Tela de visualização de imagens do Dicoogle



Fonte: Próprio Autor

Ao utilizar o Dicoogle como servidor de banco de dados de imagens médicas, a melhor opção a ser feita, é utilizar um visualizador de terceiros acoplado ao sistema, para que o usuário consiga mais opções de pós-processamento de imagens.

5.7 Possibilidades de Customização

Nesta seção serão avaliadas as possibilidades de customização dos softwares, será abordada a existência e o funcionamento dos *plugins*, bem como a possibilidade de tradução dos mesmos, e se existem opções de diagnóstico automático, ou auxílio ao diagnóstico.

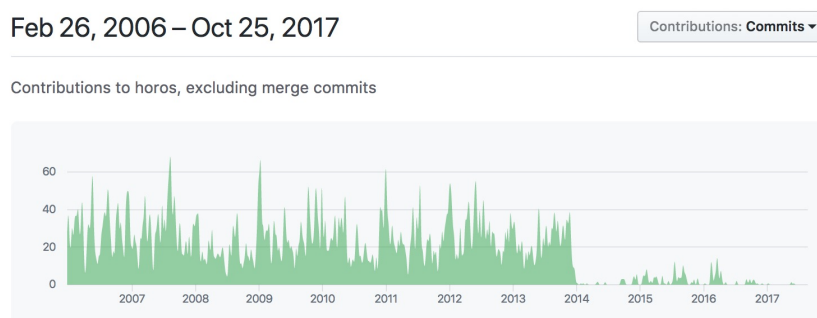
5.7.1 Horos

Em seu site oficial, o Projeto Horos instiga qualquer pessoa a participar do projeto, seja contribuindo com código, ou criando issues e pull requests no Github do projeto. E

Horos (2017) ainda destaca os tipos diferentes de usuários da comunidade open-source que podem contribuir com o projeto:

- **Proprietário** - É o usuário ou empresa que criou o projeto e tem o projeto em sua conta.
- **Mantenedores e Colaboradores** - São os usuários que efetivamente trabalham no projeto, e dão a direção a ser seguida no desenvolvimento.
- **Contribuinte** - É qualquer um que tenha tido um *pull request* aceito no projeto.
- **Membros da Comunidade** - São usuários que usam e se importam profundamente sobre o projeto, e são ativos em discussões sobre características do produto.

Figura 25: Gráfico de Commits do projeto Horos no Github

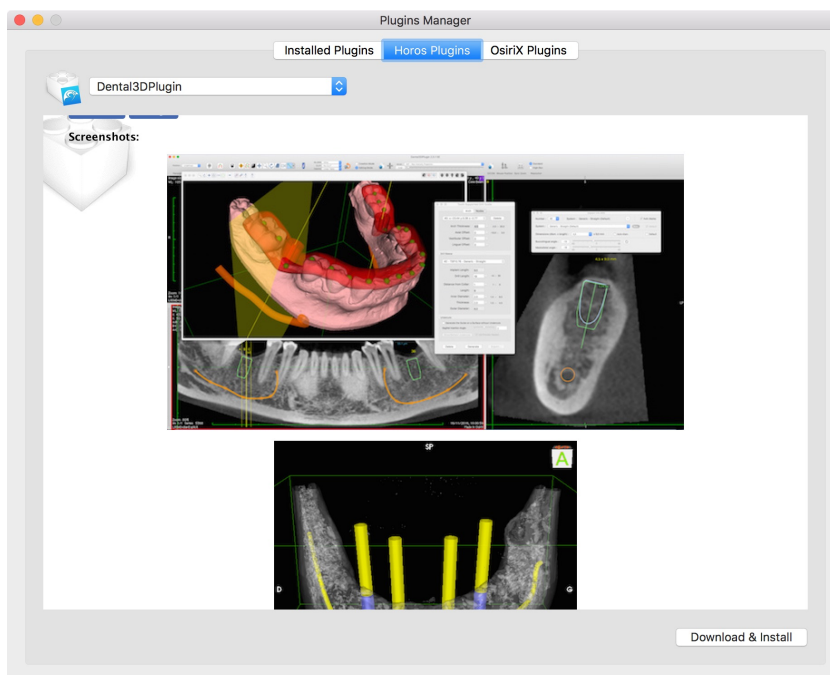


Fonte: Próprio Autor

Percebe-se na figura 25 que houveram muitos *commits* no repositório desde o seu início, em fevereiro de 2006, até março de 2014, quando o projeto deu uma arrefecida no seu desenvolvimento, porém continua recebendo atualizações esporádicas até hoje, com usuários criando discussões e solicitações de melhorias e correções de bugs.

Além do desenvolvimento do núcleo do software em si, o Horos permite a criação e utilização de *plugins* desenvolvidos pela comunidade, e também possui um repositório exclusivo para o desenvolvimento de novos. O repositório de *plugins* é menos ativo que o projeto principal, mas ainda assim recebe commits esporádicos. Na figura 26 se vê a tela de escolha e instalação de plugins no Horos.

Figura 26: Tela de escolha de plugins do Horos



Fonte: Próprio Autor

O Horos não possui tradução que venha instalada por padrão, este é um ponto que poderia ser melhorado pelos desenvolvedores do projeto, visto que o software em que é baseado, o OsiriX, possui traduções que vêm instaladas juntamente na instalação.

5.7.2 Orthanc

A comunidade é encorajada a trabalhar em conjunto para desenvolver algumas soluções que não existem no projeto original. Segundo Orthanc, (2017), núcleo do Orthanc pode ser customizado utilizando *plugins*. Os *plugins* utilizam a forma de uma biblioteca compartilhada (.dll no Windows, .so no GNU/Linux, .dylib no Mac OSX), e podem ser utilizados para várias funções, entre elas:

- Servir novas aplicações web, com acesso completo a *API REST* disponível, o que torna fácil a manipulação de imagens DICOM utilizando código JavaScript.
- Modificar a maneira como as imagens DICOM são decodificadas, um dos *plugins* que já vem no instalador padrão faz isso, a decodificação de imagens JPEG2000, opção que não está disponível no core do Orthanc.
- Alterar o banco de dados padrão no *backend* de SQLite para outro: PostgreSQL, MySQL, SQL Server, entre outros.

- Criar novas *REST APIs*, se utilizando da API padrão do Orthanc, como o *plugin* DICOMweb faz para extender o servidor para utilizar o protocolo de comunicação DICOMweb.
- Processar imagens assim que forem processadas pelo Orthanc, de maneira a executar tarefas automatizadas sem a necessidade de interação do usuário.

A tradução do Orthanc não é implementada no sistema de maneira nativa, hoje para traduzir o software é necessário compilar os arquivos binários a partir do código fonte traduzido. O autor não disponibiliza essa funcionalidade no sistema, mas prevê uma forma de funcionamento de um *plugin*, que sobrescreva os arquivos *explorer.html* e *explorer.js*.

5.7.3 Dicoogle

O Departamento de BioInformática da Universidade de Aveiro, que é responsável pelo desenvolvimento do Dicoogle, disponibiliza em sua página no GitHub um tutorial de exemplo de criação de um plugin básico, e um web service que cria uma API web para ser usada com o sistema. A documentação é simples e rasa, dando um panorama geral sobre o funcionamento dessas utilidades, mas já serve como base para um começo de desenvolvimento. Na figura 27 pode ser visto um gráfico com os *commits* desde o início do repositório em Dezembro de 2014, até hoje.

Há a possibilidade de integração com serviços de terceiros através de *plugins* que são acessíveis na pasta *Plugins*, dentro do diretório do Dicoogle. Caso não haja configuração pré-definida para os *plugins*, um novo arquivo de configurações é criado com as configurações padrões.

Figura 27: Gráfico de Commits do projeto Dicoogle no Github



Fonte: Próprio Autor

Não há uma maneira de traduzir o Dicoogle facilmente, para que isso possa ser feito, o código fonte precisa ser alterado, traduzido e compilado novamente. Durante

a pesquisa, não foram localizados *plugins* de diagnóstico automático, mas é possível desenvolver seus *plugins* para qualquer finalidade.

5.8 Dificuldades Encontradas

Não foi possível realizar a análise de dois dos softwares que foram propostos, pois apresentaram diversos problemas de instalação. E em nenhum dos casos foi possível resolver. Primeiramente, os softwares foram testados em Windows XP, e como apresentaram erros desconhecidos, foi executada a instalação dos mesmos no Windows 7, e mesmo com um sistema operacional mais moderno, não foi possível executar nenhum dos dois.

Ainda como opção de teste, foi executada a instalação dos Javas 6, 7 e 8 como alternativa para a execução dos softwares, o que não resultou em sucesso. A falta de documentação dos dois PACS e a dificuldade de contato com os autores foi fator determinante para a não realização dos testes propostos.

Abaixo, são listadas, de forma sintetizada, as dificuldades encontradas em cada um dos softwares:

5.8.1 Dificuldades com o dcm4chee

1. Dificuldade de instalação
2. Documentação confusa e desestruturada

5.8.2 Dificuldades com o Conquest

1. Mensagens de erros sem tratamento
2. Dificuldade de instalação
3. Falta documentação avançada para *debug* de erros

Quadro 5: Síntese das características analisadas nos PACS

Características	Horos	Orthanc	Dicoogle
Documentação	<ul style="list-style-type: none"> - Pouca documentação - Suporte técnico pago 	<ul style="list-style-type: none"> - Documentação bem completa - Possui área específica no site - Boa documentação sobre API REST 	<ul style="list-style-type: none"> - Pouca documentação - Não há área específica no site
Instalação	<ul style="list-style-type: none"> - Simples- Instalador - Necessita cadastro no site - Processo roda em 64 bits 	<ul style="list-style-type: none"> - Instalador Simples - Serviço do Windows - Acesso via browser - REST API - Configurações em JSON - Usuários em arquivo txt 	<ul style="list-style-type: none"> - Download em arquivo ZIP - Não requer instalação - Roda com .bat - Configurações em XML - Usuários e senhas em arquivo txt, e senha salva em hash
Armazenamento	<ul style="list-style-type: none"> - Filesystem - Opção de pasta - Mais de um diretório - Cria índice em arquivo .sql - Processo de backup próprio - Possui compressão de imagens 	<ul style="list-style-type: none"> - Padrão (filesystem) - Pasta própria - Plugins para bancos de dados - Salva JSON com DICOM para facilitar indexação - Compressão de Imagens - Salva em vários diretórios - Backup via filesystem 	<ul style="list-style-type: none"> - Filesystem - Pasta Original - Indexação - Apache Lucene - Difícil backup (devido ao controle manual de pastas) - Sem compressão
Utilização	<ul style="list-style-type: none"> - Interface nativa - Interface completa, com pré-visualização - Importação de exames ágil - Opção de busca 	<ul style="list-style-type: none"> - Interface WEB - Importação de exames via web - Opção de busca fácil de usar - Interface um pouco poluída 	<ul style="list-style-type: none"> - Não há "importação" de exames, apenas uma indexação - Navegação simples pela interface - Opção de busca ágil
Comunicação	<ul style="list-style-type: none"> - Servidor WEB - Listener TCP para comunicação com outras máquinas na rede - Comunicação através do padrão DICOM - Comunicação entre servidores 	<ul style="list-style-type: none"> - Maior vantagem é a API REST - Toda a interface web utiliza a API REST - API com documentação completa - Possibilidade de extensão - Comunicação entre servidores padrão DICOM 	<ul style="list-style-type: none"> - Possui API WEB - API não possui muita documentação - Comunicação através do protocolo DICOM - Comunicação entre servidores
Visualização	<ul style="list-style-type: none"> - Opção mais completa - Visualizador com diversas opções úteis - Gerador de imagens 3D - Compatível com OpenGL - Ótima performance 	<ul style="list-style-type: none"> - Poucas opções de pós processamento de imagens (contraste, brilho, zoom) - Básicas porém funcionais 	<ul style="list-style-type: none"> - Visualização via web - Sem opções de visualização - Visualização simples - Não possui zoom, contraste ou brilho
Customização	<ul style="list-style-type: none"> - Alterações através de Issues e pull requests - Mais atualizações de 2006 a 2014 - Plugins - Não possui tradução 	<ul style="list-style-type: none"> - API REST - Comunidade - Escolha de banco de dados - Plugins - Automatização de processos - Não possui tradução 	<ul style="list-style-type: none"> - Plugins - Exemplo Github - Não há tradução - Software precisa ser recompilado para traduzir

Fonte: Próprio Autor

6 Conclusão

Neste trabalho pode-se conferir uma introdução sobre a utilização de sistemas PACS. Foi abordado o funcionamento destes sistemas, e as vantagens de uso nos ambientes de diagnóstico. Elencou-se os PACS open-source que foram avaliados neste trabalho e avaliados conforme os critérios descritos no Capítulo 4.1.

A presente análise teve como problemática, a necessidade de clínicas e hospitais em realizarem a visualização e armazenamento de exames médicos em imagens de maneira ágil e eficiente, para auxiliar na tomada de decisão e diagnóstico rápido do paciente. Para tal análise, foi simulada uma utilização mais prática do uso dos PACS em um ambiente controlado, onde foram simuladas operações padrões, tais como a importação e aquisição de exames, busca de pacientes e visualização de imagens.

Conforme mencionado no capítulo 1, a implantação de um ambiente *filmless*, resulta em grandes avanços operacionais, e é importante que o sistema PACS utilizado atenda aos requisitos necessários, para prover um atendimento de excelência para que o atendimento aos pacientes seja excelente. Nesse sentido, os profissionais que utilizam estes softwares conseguem realizar o seu trabalho de forma mais rápida e eficiente.

A meta inicial deste trabalho era realizar uma comparação entre cinco softwares de bancos de dados de imagens médicas, mas com os problemas encontrados para realizar os testes nos softwares Conquest e dcm4chee, foram analisados apenas três, o Orthanc, o Dicoogle e o Horos.

Com os PACS instalados, foi utilizado o *workflow* descrito no Capítulo 4.1.2, e os testes foram executados igualmente em cada um dos softwares, dentro das opções disponíveis em cada programa.

Dos PACS analisados neste trabalho, os que se saíram melhores na avaliação foram o Orthanc e o Horos. O primeiro, se destaca na possibilidade de customização e pela sua robusta API REST, que facilita o trabalho de desenvolvedores terceiros aos projetos para utilizarem o Orthanc como base nas suas aplicações. A utilização do Orthanc como servidor possibilita a utilização da API Rest que já vem embutida em seu core, que permite que sejam desenvolvidos novas soluções, principalmente web, utilizando como base este recurso do servidor.

Já o Horos, apesar de ser apenas utilizável em computadores com Mac OS, apresentou uma qualidade superior aos outros, nele foram encontradas opções muito avançadas em relação aos PACS analisados anteriormente, ferramentas que são muito úteis aos profissionais da área de diagnóstico por imagem, e ainda possui a opção de adicionar *plugins* de terceiros ao programa, o que acaba abrindo mais o leque de possibilidades disponíveis para visualização e diagnóstico de imagens médicas. Com suporte

a aceleração gráfica via hardware, o Horos funciona muito bem como cliente de visualização, e pode ser utilizado em conjunto com outros servidores, como o Orthanc, por exemplo.

Dados os resultados obtidos nesta análise, a utilização do Orthanc como servidor PACS se torna uma opção robusta, devido à facilidade de configuração e colocação em uso de um ambiente pronto para a utilização de diversos visualizadores diferentes na rede e a possibilidade de comunicação entre máquinas de aquisição de exames e outros servidores. Já em ambientes que possuem *hosts* com Mac OS na rede, o Horos se mostra muito mais desenvolvido na visualização de imagens, e com muitas opções a mais que os concorrentes.

Este trabalho pode servir de base para outras pesquisas, ou como fonte de informação para alguma clínica ou instituição que esteja com dúvidas para a escolha de um PACS que tenha as funcionalidades desejadas pela sua empresa.

6.1 Trabalhos Futuros

Para trabalhos futuros, é interessante dedicar um estudo maior ao sigilo dos dados, e as configurações de acesso aos exames cadastrados nos PACS analisados anteriormente, visto que a segurança de informação no ramo médico é muito importante, de maneira a garantir o sigilo de informações pessoais dos pacientes.

REFERÊNCIAS BIBLIOGRÁFICAS

- ALVES, A. F. P. (2016). Dicoogle: No-sql for supporting big data environments. Master's thesis, Universidade de Aveiro.
- BIANCHI, L. (2017). **Qualidade de produtos de software**. <http://bianchi.pro.br/edutec/qualsoft.php>. (Accessado em 24/05/2017).
- CARITÁ, E. C., MATOS, A. L. M., and AZEVEDO-MARQUES, P. M. d. (2004). **Feramentas para visualização de imagens médicas em hospital universitário**. *Radiol Bras*, 37:437–40.
- CONQUEST (2017). **Conquest DICOM Software**. <https://ingenium.home.xs4all.nl/dicom.html>. Accessado em: 2017-04-12.
- COSTA, C., FERREIRA, C., BASTIÃO, L., RIBEIRO, L., SILVA, A., and OLIVEIRA, J. L. (2011). **Dicoogle-an open source peer-to-peer PACS**. *Journal of digital imaging*, 24(5):848–856.
- DCM4CHE (2017). **Open Source Clinical Image and Object Management**. <http://www.dcm4che.org/>. Accessado em: 2017-04-12.
- DE AZEVEDO-MARQUES, P. M. and SALOMÃO, S. C. (2009). **PACS: sistemas de arquivamento e distribuição de imagens**. *Revista Brasileira de Física Médica*, 3:131–139.
- DIAS, L. G. and DA COSTA, V. G. (2017). Gestão, formalismos e desenvolvimento de software-uma análise acerca das produções brasileiras. *Blucher Education Proceedings*, 2(1):88–95.
- DICOOGLE (2017). **DICoogle**. <http://www.dicoogle.com/>. Accessado em: 2017-04-12.
- DOCKER (2017). **Docker Documentation**. <https://docs.docker.com/get-started/>. (Accessado em 08/08/2017).
- DREYER, K. J., HIRSCHHORN, D., THRALL, J. H., and PACS, M. (2006). **A guide to the digital revolution**. Springer.
- EMER, M. C. F. P. (2017). **Conceitos Básicos sobre Qualidade**. <http://www.inf.ufpr.br/silvia/ES/qualidade/produtos.pdf>. (Accessado em 06/04/2017).

- FIRMINO FILHO, J. M., VALENTIM, R., RIBEIRO, M., and CAVALCANTI, L. (2013). **OpenPACS—Sistema Open Source para Comunicação e Arquivamento de Imagens Médicas: Relato de Experiência em um Hospital Universitário**. *Revista Eletrônica de Comunicação, Informação & Inovação em Saúde*, 7(2).
- GLADCHEFF, A. P. (2001). *Um instrumento de avaliação da qualidade para software educacional de matemática*. PhD thesis, Universidade de São Paulo.
- Guimarães, R. R. (2002). *Conversão de Imagens do Formato DICOM Visando a Inter-Operacionalidade de Sistemas Através da WEB*. PhD thesis, UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL.
- HOROS (2017). **Horos or OsiriX: What are the differences?** <https://www.horosproject.org/horos-or-osirix/>. Acessado em: 2017-04-12.
- ISO/IEC (2001). *NBR ISO/IEC 9126-1: Engenharia de software - Qualidade de Produto - Parte 1: Modelo de qualidade*. ISO/IEC.
- Jodogne, S., Bernard, C., Devillers, M., Lenaerts, E., and Coucke, P. (2013). **Orthanc—A lightweight, restful DICOM server for healthcare and medical research**. In *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, pages 190–193. IEEE.
- MULLOY, B. (2012). *Web API Design: Crafting Interfaces that Developers Love*. ApiGee. <https://pages.apigee.com/web-api-design-website-h-ebook-registration.html>.
- NDT RESOURCE CENTER, I. S. U. (2013). **History of Radiography**. <https://www.nde-ed.org/EducationResources/CommunityCollege/Radiography/Introduction/hist>. Acessado em: 2017-05-22.
- OPENGL (2017). **OpenGL**. <http://opengl.org>. Acessado em: 2017-08-21.
- ORTHANC (2017). **Orthanc – DICOM Server**. <http://www.orthanc-server.com/>. Acessado em: 2017-04-12.
- OSIRIX (2017). **OsiriX — PACS**. <http://www.osirix-viewer.com/resources/pacs/>. Acessado em: 2017-04-12.
- PRODANOV, C. C. and DE FREITAS, E. C. (2013). *Metodologia do Trabalho Científico: Métodos e Técnicas da Pesquisa e do Trabalho Acadêmico-2ª Edição*. Editora Feevale.
- QUICKTIME (2017). **Quicktime**. <http://www.quicktime.com>. Acessado em: 2017-09-15.

- RATIB, O., RODUIT, N., NIDUP, D., DE GEER, G., ROSSET, A., and GEISSBUHLER, A. (2016). **PACS for Bhutan: a cost effective open source architecture for emerging countries.** *Insights into Imaging*, 7(5):747–753.
- ROSSET, A., SPADOLA, L., and RATIB, O. (2004). **OsiriX: an open-source software for navigating in multidimensional DICOM images.** *Journal of digital imaging*, 17(3):205–216.
- SEIXAS, F. L. (2005). **Diagnóstico Auxiliado por Computador.** In *Monografia*. Departamento em Engenharia da Computação, Escola de Engenharia, Universidade Federal Fluminense.
- SIEGEL, E. and REINER, B. (2002). **Work flow redesign: the key to success when using PACS.** *American Journal of Roentgenology*, 178(3):563–566.
- SILVA, D. C. d. (2015). **Exames por imagem no diagnóstico de anquilose alveolodentária: relato de caso clínico.**
- STRICKLAND, N. H. (2000). **PACS (picture archiving and communication systems): filmless radiology.** *Archives of disease in childhood*, 83(1):82–86.
- VALENTE, F., SILVA, L. A. B., GODINHO, T. M., and COSTA, C. (2016). **Anatomy of an extensible open source PACS.** *Journal of digital imaging*, 29(3):284–296.
- VTK (2017). **The Visualization Toolkit (VTK).** <https://public.kitware.com/>. Acessado em: 2017-08-22.