

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**IMPLEMENTAÇÃO DE UM MÓDULO
DE IMPRESSÃO DE IMAGENS
DIAGNÓSTICAS EM UM
VISUALIZADOR BASEADO EM *WEB***

TRABALHO DE GRADUAÇÃO

Marcelo Arrais Porto

Santa Maria, RS, Brasil

2012

IMPLEMENTAÇÃO DE UM MÓDULO DE IMPRESSÃO DE IMAGENS DIAGNÓSTICAS EM UM VISUALIZADOR BASEADO EM *WEB*

por

Marcelo Arrais Porto

Trabalho de Graduação apresentado ao Curso de Ciência da Computação
da Universidade Federal de Santa Maria (UFSM, RS), como requisito
parcial para a obtenção do grau de
Bacharel em Ciência da Computação

Orientador: Prof. Dr. Marcos Cordeiro d'Ornellas

Co-orientador: MSc. Jean Carlo Albiero Berni

Trabalho de Graduação N. 343

Santa Maria, RS, Brasil

2012

**Universidade Federal de Santa Maria
Centro de Tecnologia
Curso de Ciência da Computação**

A Comissão Examinadora, abaixo assinada,
aprova o Trabalho de Graduação

**IMPLEMENTAÇÃO DE UM MÓDULO DE IMPRESSÃO DE
IMAGENS DIAGNÓSTICAS EM UM VISUALIZADOR BASEADO
EM *WEB***

elaborado por
Marcelo Arrais Porto

como requisito parcial para obtenção do grau de
Bacharel em Ciência da Computação

COMISSÃO EXAMINADORA:

Prof. Dr. Marcos Cordeiro d'Ornellas
(Presidente/Orientador)

Prof. Dr. Cesar Tadeu Pozzer (UFSM)

Prof. Dr. José Antônio Trindade Borges da Costa (UFSM)

Santa Maria, 02 de Julho de 2012.

AGRADECIMENTOS

Agradeço a Deus e à minha família, especialmente aos meus pais, pelo apoio e incentivo fornecidos. Aos amigos e colegas da Animati Computação Aplicada: Jean (que também foi meu co-orientador neste trabalho), Cesar, Cristiano e Gabriela, os quais me proporcionaram uma grande aquisição de conhecimento nos últimos dois anos. Ao professor Marcos, meu orientador, por ter me dado um direcionamento profissional no meu segundo ano de curso, pois foi através dele que tive meu primeiro contato com a Animati e minha primeira oportunidade de trabalhar com aplicações reais. Ao Nicolas Roduit que, mesmo distante, me auxiliou durante todo o processo de implementação deste trabalho. E por último, mas não menos importante, agradeço a todos os professores e colegas que fizeram parte desta caminhada nesses quatro anos e meio de curso, em especial ao pessoal da turma de 2008, parceiros de estudos, churrascos, partidas de truco e junções nerds para jogar video game. Muito obrigado a todos vocês.

*“Run, rabbit, run.
Dig that hole, forget the sun,
And when at last the work is done
Don’t sit down it’s time to dig another one.”*
— ROGER WATERS

RESUMO

Trabalho de Graduação
Curso de Ciência da Computação
Universidade Federal de Santa Maria

IMPLEMENTAÇÃO DE UM MÓDULO DE IMPRESSÃO DE IMAGENS DIAGNÓSTICAS EM UM VISUALIZADOR BASEADO EM *WEB*

Autor: Marcelo Arrais Porto

Orientador: Prof. Dr. Marcos Cordeiro d'Ornellas

Co-orientador: MSc. Jean Carlo Albiero Berni

Local e data da defesa: Santa Maria, 02 de Julho de 2012.

A Medicina é uma área de conhecimento que se beneficiou com a popularização dos computadores. A criação e o desenvolvimento da Radiologia digital, bem como o surgimento de padrões como o DICOM (Digital Imaging and Communications in Medicine), trouxeram maior rapidez e precisão na realização de exames, características positivas tanto para médicos quanto para pacientes. Ainda assim, é possível observar que exames impressos em papel continuam sendo necessários para que possam ser entregues a qualquer paciente e transportados de maneira prática para fora da clínica. Este trabalho tem como objetivo acrescentar a funcionalidade de impressão, tanto em papel comum quanto em filmes radiográficos, a um visualizador de imagens diagnósticas baseado em web chamado Weasis.

Palavras-chave: Impressão; DICOM; Weasis; imagens diagnósticas.

ABSTRACT

Trabalho de Graduação
Curso de Ciência da Computação
Universidade Federal de Santa Maria

IMPLEMENTATION OF A MEDICAL IMAGES PRINTING MODULE IN A WEB-BASED VIEWER

Author: Marcelo Arrais Porto
Advisor: Prof. Dr. Marcos Cordeiro d'Ornellas
Coadvisor: MSc. Jean Carlo Albiero Berni

Medicine is a field which benefited from the popularization of computers. The creation and expansion of Digital Radiology, as well as the appearance of standards such as DICOM (Digital Imaging and Communications in Medicine), brought more fastness and precision to perform examinations, positive features to physicians and patients. Nevertheless it is possible to notice that the printing of exams in paper is still needed, so they can be delivered to any kind of patient and carried outside the clinic in an easy way. This study aims to add the print feature in paper and radiographic film to Weasis, a web-based viewer of medical images.

Keywords: printing, DICOM, Weasis, medical images.

LISTA DE FIGURAS

2.1	Modelo real em comparação ao modelo DICOM	19
2.2	Interface do Weasis com exemplos de medições nas imagens	20
3.1	Representação do processo de impressão de imagens diagnósticas	22
3.2	Arquitetura dos módulos do Weasis	28
3.3	Janela da interface gráfica para impressão comum	29
3.4	Diagrama de classes com as classes desenvolvidas e utilizadas no processo de impressão comum	30
3.5	Tomografia do tipo QC utilizada na classe-teste da impressão DICOM	31
3.6	Tela inicial do PrintServerDummy para ajustes de configuração.....	32
3.7	Emulador SCP do Dicom Validation Tool	33
3.8	Janela da interface gráfica para impressão DICOM.....	36
3.9	Classes criadas no Weasis referentes à impressão DICOM.....	38
3.10	Representação da impressão comum do layout 4x4 do Weasis em uma folha A4	40

LISTA DE TABELAS

3.1	Tabela de atributos da Basic Film Session SOP Class para uma aplicação SCU (NEMA, 2008).....	24
3.2	Tabela de atributos da Basic Film Box SOP Class para uma aplicação SCU (NEMA, 2008).....	25
3.3	Tabela de atributos da Basic Grayscale Image Box SOP Class para uma aplicação SCU (NEMA, 2008).	26

LISTA DE CÓDIGOS

3.3.1 Armazenando um arquivo DICOM em um DicomObject.	33
3.3.2 Definição e configuração das AEs.	34
3.3.3 Resposta da mensagem N-CREATE da Film Box é armazenada para termos acesso à Image Box criada.	35
3.4.1 Método que realiza a conversão de uma imagem colorida em uma imagem monocromática no Weasis	37
3.4.2 Trecho de código em que os atributos do objeto DICOM recebem valores diferentes conforme o tipo de impressão selecionado (tons de cinza ou colorida)	39

LISTA DE ABREVIATURAS E SIGLAS

ACR	American College of Radiology
AE	Application Entity
CR	Computed Radiography
CT	Computed Tomography
DICOM	Digital Imaging and Communications in Medicine
DIMSE	DICOM Message Service Element
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IOD	Information Object Definition
NEMA	National Electrical Manufacturers Association
PACS	Picture Archiving and Communication System
QC	Quality Control
SCP	Service Class Provider
SCU	Service Class User
SOP	Service-Object Pair
TS	Transfer Syntax
UID	Unique Identifier
VR	Value Representation
WADO	Web Access to Dicom Objects
XML	Extensible Markup Language

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Objetivos	15
1.1.1	Objetivo geral	15
1.1.2	Objetivos específicos	15
1.2	Estrutura do Trabalho	15
2	REVISÃO DE CONCEITOS	16
2.1	DICOM	16
2.1.1	PACS	17
2.1.2	Application Entities e Service-Object Pairs	17
2.1.3	Unique Identifiers	17
2.1.4	Abstract Syntax e Transfer Syntax	17
2.1.5	DIMSE	18
2.1.6	Modelo básico	18
2.2	dcm4che	19
2.2.1	Weasis	19
3	DESENVOLVIMENTO E METODOLOGIA UTILIZADA	21
3.1	Processo de impressão de imagens diagnósticas	21
3.1.1	Impressão comum	22
3.1.2	Impressão DICOM	23
3.2	Implementação do recurso de impressão de imagens diagnósticas em impressoras comuns	27
3.2.1	Estratégia e estudo sobre o Weasis	27
3.2.2	Codificação	27
3.3	Implementação do recurso de impressão de imagens diagnósticas em impressoras DICOM	29
3.3.1	Estratégia	29
3.3.2	Escolha de emuladores Print SCP	31
3.3.3	Codificação	33
3.4	Integração da impressão DICOM com o Weasis	35
3.4.1	Adaptação do código desenvolvido inicialmente	35
3.4.2	Adaptação do código para impressão DICOM colorida	37
3.5	Testes	37
3.6	Resultados	39
4	CONCLUSÃO E TRABALHOS FUTUROS	42

REFERÊNCIAS	44
--------------------------	-----------

1 INTRODUÇÃO

Ao longo dos anos, a evolução da Computação e da tecnologia de modo geral têm contribuído no avanço das mais diversas áreas de conhecimento. Os formatos digitais de informação estão cada vez mais presentes no dia-a-dia de profissionais como os da área da saúde, que precisam estar constantemente atualizados em relação às tecnologias utilizadas nesse meio. A Radiologia é um ramo da Medicina que foi impactado pelos avanços dos computadores.

Segundo BLOCKER (2010), apesar do raio-x ter sido descoberto no final do século XIX, a Radiologia digital pouco avançou até a década de 1970, quando começou a se popularizar e ser implantada em clínicas e hospitais de grande e pequeno porte, justamente na mesma época em que os computadores pessoais estavam se tornando mais acessíveis e presentes na vida das pessoas. A partir daí, descobriu-se que as imagens de raio-x podiam ser armazenadas e reproduzidas em telas de fósforo, ideia que serve como base para os sistemas de CR (Computed Radiography).

Para possibilitar a visualização e manipulação dessas imagens geradas por equipamentos de radiologia digital, surgiram as chamadas *workstations*. Um exemplo dentre essas ferramentas é o projeto Weasis (RODUIT, 2010), um visualizador de imagens diagnósticas de código aberto desenvolvido na linguagem Java. O Weasis faz parte da iniciativa dcm4che, que é um conjunto de aplicativos open source direcionados para o segmento da saúde e que tem recebido aceitação mundial com diversos casos de sucesso implementados (EVANS et al., 2012).

Mesmo considerando que a distribuição digital das imagens diagnósticas tenha se popularizado, em alguns casos ainda existe a necessidade da impressão dos exames para que sejam entregues aos pacientes e/ou médicos solicitantes. O processo de impressão desse tipo de imagem é complexo por envolver fatores como a comunicação realizada

entre equipamentos da área médica e o fato de que informações relevantes não podem ser perdidas. Espera-se que uma radiografia seja uma representação fiel do paciente para que o médico seja capaz de identificar enfermidades e alterações físicas e, dessa forma, realizar um diagnóstico preciso.

Com isso, percebeu-se a necessidade de agregar essa funcionalidade ao Weasis, pois o visualizador, apesar de disponibilizar diversas funcionalidades para a manipulação de imagens médicas, ainda não contava com o recurso de impressão das mesmas até o início deste trabalho.

1.1 Objetivos

1.1.1 Objetivo geral

O objetivo geral deste trabalho é adicionar à *workstation* Weasis o recurso de impressão de imagens em papel e filmes radiográficos para que clínicas e profissionais da saúde que utilizam o visualizador possam realizar essa tarefa através da aplicação.

1.1.2 Objetivos específicos

- Imprimir imagens em impressoras comuns através do Weasis;
- Imprimir imagens em impressoras de imagens diagnósticas através do Weasis;
- Elaborar uma interface de fácil manipulação para os usuários do Weasis a fim de que possam fazer uso da funcionalidade.

1.2 Estrutura do Trabalho

O texto deste trabalho está estruturado da seguinte maneira: no capítulo 2, é feita uma revisão de conceitos a partir da fundamentação teórica, a qual é necessária para uma melhor compreensão do trabalho realizado. O capítulo 3 aborda a metodologia utilizada e o desenvolvimento como um todo, incluindo a estratégia de implementação e a parte da codificação, bem como os testes realizados e os resultados que foram obtidos. No capítulo 4, é feita a conclusão do trabalho e a apresentação de ideias, relacionadas ao assunto abordado, que podem ser desenvolvidas no futuro.

2 REVISÃO DE CONCEITOS

Este capítulo aborda conceitos relativos à manipulação de imagens diagnósticas digitais, incluindo o padrão DICOM (Digital Imaging and Communications in Medicine) e ferramentas utilizadas no desenvolvimento do trabalho, os quais são necessários para a compreensão do recurso implementado.

2.1 DICOM

O surgimento dos equipamentos digitais de imagens diagnósticas foi acompanhado pela dificuldade de comunicá-los entre si. Cada fabricante desenvolvia um padrão próprio de comunicação para seus aparelhos, bem como um formato diferente de armazenamento para os dados que eram gerados, resultando em diversos formatos proprietários de arquivos. Essas diferenças técnicas entre formatos representavam um problema para que informações de pacientes fossem compartilhadas entre clínicas, pois isso só seria possível se ambas possuíssem equipamentos do mesmo fabricante.

Para solucionar essa questão, a NEMA (National Electrical Manufacturers Association) e a ACR (American College of Radiology) desenvolveram o DICOM, um padrão, independente de fabricante ou modelo, com o objetivo de aumentar a interoperabilidade técnica entre equipamentos de imagens diagnósticas. Por volta de 2004, onze anos após o lançamento inicial do DICOM, a maioria dos dispositivos que realizavam ultrassom, CT (Computed Tomography) e CR já haviam adotado o padrão (HOWARD, 2011).

Todas as informações médicas são agrupadas em objetos DICOM, os quais possuem propriedades e atributos. As informações particulares de cada objeto são descritas pelas chamadas IODs (Information Object Definitions). Por exemplo, pode-se considerar um paciente cujo IOD é descrito pelo seu nome, sexo, idade, altura e outros atributos que possam ser relevantes. Existem mais de 2000 atributos dentro do DICOM, os quais compõem

uma lista chamada DICOM Data Dictionary (PIANYKH, 2008).

2.1.1 PACS

De acordo com PIANYKH (2008), o conceito de PACS (Picture Archiving and Communication System) está intrinsecamente relacionado ao DICOM. Esses sistemas (constituídos por *hardware* e *software*), baseados no padrão, são voltados para o armazenamento e manipulação de imagens diagnósticas digitais. Os PACS abrangem dispositivos geradores de imagens (como tomógrafos, por exemplo), arquivos (depósitos) de imagens (onde elas são armazenadas) e *workstations*. Cada PACS é acompanhado por seu próprio DICOM Conformance Statement, um documento que informa como e quanto o sistema está de acordo com o padrão.

2.1.2 Application Entities e Service-Object Pairs

Após as informações serem capturadas como atributos DICOM, elas podem ser manipuladas e transferidas entre os dispositivos e *softwares* que fazem uso do padrão, os quais são chamados de AEs (Application Entities). As transferências de dados entre aplicações DICOM envolvem serviços, como o de impressão, por exemplo, os quais são associados a determinados tipos de objetos DICOM. Essa associação resulta nos chamados SOPs (Service-Object Pairs), que o padrão agrupa em classes conhecidas como SOP Classes.

2.1.3 Unique Identifiers

Um UID ou UI (Unique Identifier) é um número que identifica determinadas instâncias de um objeto DICOM. Por exemplo, todo exame ou SOP Class possui um UID que os identifica. Esse número deve ser único em um escopo global, ou seja, se copiarmos um exame de uma unidade DICOM para outra, a segunda deve modificar o valor do Study UID do objeto para que as alterações que venham a ser realizadas nele não se confundam com as propriedades do original (PIANYKH, 2008).

2.1.4 Abstract Syntax e Transfer Syntax

Abstract Syntaxes descrevem serviços que AEs podem prover ou utilizar, como a impressão ou o armazenamento (*storage*) de imagens. Uma Abstract Syntax pode ser vista como um codenome para funções do DICOM que envolvem comunicação em rede.

A Transfer Syntax é o que possibilita a interação entre *hardwares* e sistemas diferen-

tes que utilizam o padrão DICOM. Enquanto uma Abstract Syntax especifica o que deve ser feito, uma Transfer Syntax é responsável por indicar como isso será feito. Resumidamente, pode-se dizer que ela explicita como as informações e as mensagens que serão transferidas estão codificadas (PIANYKH, 2008).

A associação ou negociação (termo utilizado no DICOM) entre Abstract e Transfer Syntax resulta no chamado Presentation Context. O exemplo de como isso acontece no serviço de impressão será abordado mais adiante no desenvolvimento deste trabalho.

2.1.5 DIMSE

DIMSE (DICOM Message Service Element) é o serviço de mensagens que possibilita a comunicação e a troca de informações entre AEs. Requisições são feitas pelas aplicações SCU (Service Class User) para as SCPs (Service Class Providers), sendo que o meio de comunicação entre ambas é o DIMSE.

Serviços DIMSE normalmente são associados aos prefixos "C" e "N", que respectivamente especificam se os dados contidos na mensagem são compostos ou normalizados. Uma IOD normalizada representa uma única entidade do mundo real no DICOM, como um paciente, por exemplo, enquanto uma IOD composta representa várias entidades reais em um único objeto, como é o caso de uma tomografia (CT image) cuja IOD contém não só informações do paciente, mas também do tomógrafo, entre outros detalhes (PIANYKH, 2008).

2.1.6 Modelo básico

O modelo DICOM pode ser visto como uma hierarquia de objetos. Quando um exame (também chamado de estudo) é realizado, é gerado um conjunto de imagens associado ao paciente. Esse conjunto que compõe o estudo é organizado em séries, que por sua vez são subdivididas em instâncias, no caso, as imagens propriamente ditas (Figura 2.1). Uma série normalmente agrupa instâncias de uma mesma modalidade (CT, por exemplo) ou imagens do paciente em uma mesma posição corporal.

Do ponto de vista de um modelo relacional de banco de dados, tem-se a seguinte estrutura: um paciente pode ter 1..n exames associados a ele, sendo que cada exame é constituído por 1..n séries. Cada série irá conter 1..n objetos DICOM (EVANS, 2006a).

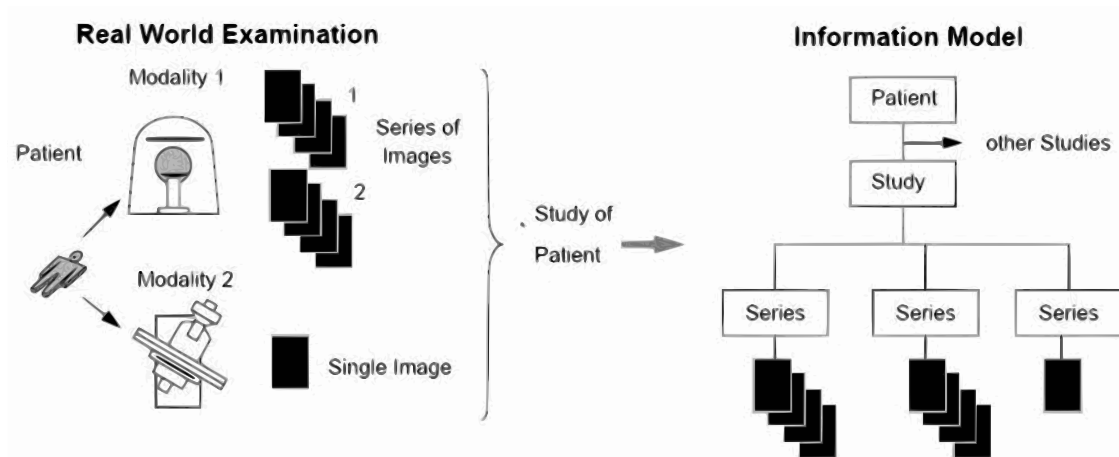


Figura 2.1: Modelo real em comparação ao modelo DICOM.

2.2 dcm4che

O *dcm4che* é um conjunto de ferramentas de código-aberto desenvolvidas em Java que implementam o padrão DICOM. Entre essas ferramentas, duas se relacionam diretamente com o assunto deste trabalho: o *dcm4che2 toolkit*, que é uma implementação do DICOM na linguagem Java, e a *workstation Weasis*.

2.2.1 Weasis

Weasis é um visualizador que foi desenvolvido na linguagem Java com o objetivo de proporcionar o acesso via web à imagens diagnósticas oferecendo recursos multimídia. Essa ferramenta é facilmente adaptável a qualquer PACS, o que pode justificar o sucesso da sua integração a sistemas comerciais (RODUI, 2010). Entre os recursos do Weasis pode-se citar a possibilidade de se realizar medições na imagem (Figura 2.2), aplicação de filtros e janelamento, a visualização de múltiplas imagens na tela, entre outros, além de oferecer suporte a WADO (Web Access to Dicom Objects). O WADO é um serviço que permite que objetos DICOM em um repositório sejam acessados através de requisições HTTP (Hypertext Transfer Protocol) ou HTTPS (Hypertext Transfer Protocol Secure).

Este trabalho é uma contribuição ao projeto Weasis, pois a implementação da impressão de imagens diagnósticas realizada será adicionada ao visualizador como uma nova funcionalidade.

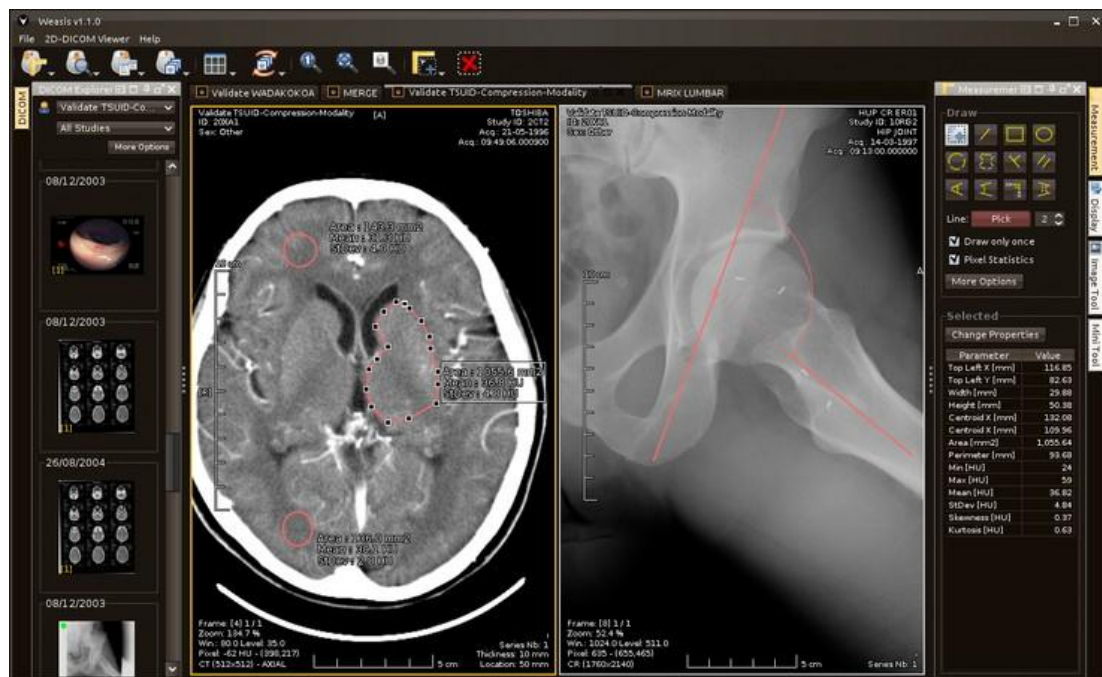


Figura 2.2: Interface do Weasis com exemplos de medições nas imagens.

3 DESENVOLVIMENTO E METODOLOGIA UTILIZADA

Neste capítulo são descritas as atividades realizadas no desenvolvimento do trabalho, juntamente com a metodologia empregada. A seção 3.1 aborda como é realizada a impressão das imagens diagnósticas, enquanto as seções 3.2 e 3.3 tratam sobre a implementação dessa ideia, incluindo a estratégia adotada e a codificação. Na seção 3.4 é relatado o processo de integração da funcionalidade com o Weasis. As seções subsequentes (3.5 e 3.6) abordam os testes realizados e os resultados obtidos ao final do trabalho.

O desenvolvimento do presente trabalho foi realizado em parceria com a empresa Animati Computação Aplicada, que possui sede na Incubadora Tecnológica de Santa Maria/RS, da qual o autor desta monografia é estagiário. A Animati tem por objetivo levar ao mercado soluções de Computação aplicada aos campos da ciência e saúde. O seu produto principal é o Animati PACS - Sistema de Arquivamento e Comunicação de Imagens Diagnósticas (ANIMATI, 2011). Assim, o desenvolvimento deste projeto vai ao encontro das expectativas da empresa no que se refere à evolução dos produtos e serviços que a mesma oferece.

3.1 Processo de impressão de imagens diagnósticas

Primeiramente foi necessária a realização de uma pesquisa sobre como imagens diagnósticas são impressas em papel comum ou em filmes radiográficos. Para isso, foi observado o funcionamento das *workstations* eFilm (versão 1.5.3) e O3-RWS, as quais já possuíam o recurso previamente implementado. A segunda ferramenta, por possuir código-aberto, possibilitou uma análise detalhada do processo.

Percebeu-se que ambos os visualizadores analisados ofereciam duas possibilidades para imprimir imagens: a impressão comum (*standard print*) e a impressão DICOM (*DICOM print*). A Figura 3.1 mostra o processo de impressão da maneira que ele é abordado

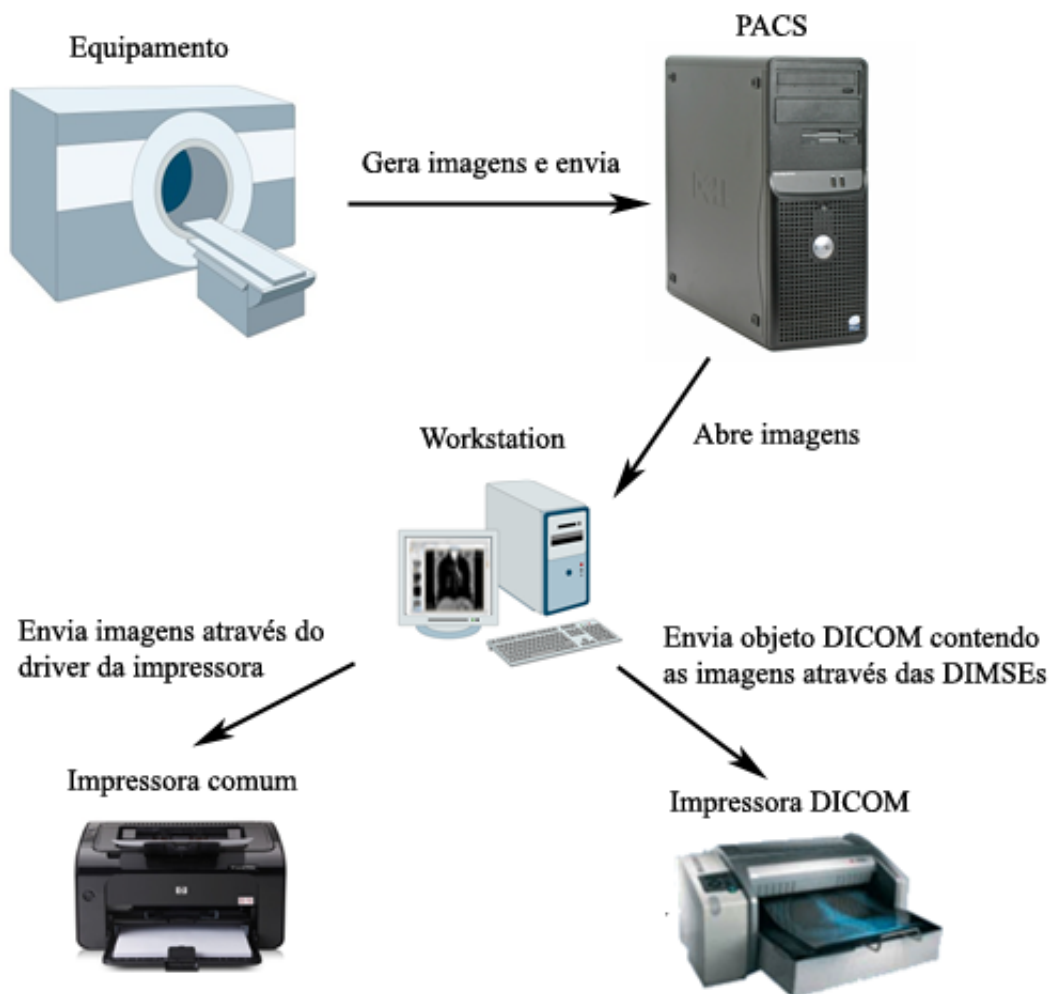


Figura 3.1: Representação do processo de impressão de imagens diagnósticas.

neste trabalho. Os dois tipos de impressão serão detalhados nas próximas seções.

3.1.1 Impressão comum

A impressão comum de imagens diagnósticas é realizada através de impressoras convencionais - a *laser*, jato de tinta ou cera - e tem importância para clínicas ou hospitais devido ao seu baixo custo se comparada à impressão em filmes, a qual é realizada em dispositivos que utilizam a tecnologia *dry laser* que será comentada mais adiante.

O procedimento faz com que os píxeis de uma imagem DICOM sejam decodificados e armazenados em um buffer, de modo que eles possam ser exibidos em um monitor convencional RGB ou enviados para impressoras que não oferecem suporte ao padrão DICOM. Após essa decodificação, o processo pode ser gerenciado pelo *driver* do dispositivo instalado no computador, o mesmo que ocorre com um documento de texto, por exemplo.

3.1.2 Impressão DICOM

A impressão DICOM é realizada em dispositivos que suportam o padrão e que normalmente fazem uso da tecnologia *dry laser*, a qual substitui os filmes revelados quimicamente e não gera resíduos poluentes (GAHLEITNER et al., 1999). O processo de impressão, por sua vez, é feita de maneira diferente em comparação à impressão comum e envolve mais detalhes. A impressora irá receber um objeto DICOM contendo a imagem sem compressão e os atributos de configuração, como o número de cópias a serem impressas, tipo e tamanho do filme radiográfico, entre outros. Esse objeto é construído dentro da aplicação que o DICOM classifica como SCU, a qual faz uso do serviço de impressão que é fornecido pelo SCP, no caso, a impressora DICOM propriamente dita.

Existem duas SOP Classes responsáveis pelo gerenciamento do processo de impressão definidas no padrão: a Basic Grayscale Print Management Meta SOP Class e a Basic Color Print Management Meta SOP Class. Como os próprios nomes sugerem, a primeira é utilizada quando se deseja imprimir uma imagem em tons de cinza, ao passo que a segunda deve ser empregada para imprimir imagens coloridas. Ambas as SOP Classes em questão, por serem tratadas de um serviço, possuem Abstract Syntax, de modo que é necessário negociar um Presentation Context, associando então a impressão a um ou mais dos três Transfer Syntax suportados pelo processo: Implicit VR Little Endian, Explicit VR Little Endian e Explicit VR Big Endian. A escolha dos TS que serão negociados deve ser feita de acordo com os que estão presentes nas imagens a serem impressas.

Depois de feita a negociação do Presentation Context, o objeto DICOM deve ser enviado ao Print SCP através de mensagens do tipo DIMSE-N. O padrão define algumas SOP Classes e atributos que obrigatoriamente devem constar em uma aplicação do tipo Print SCU:

- Basic Film Session SOP Class: uma Basic Film Session especifica que um novo processo de impressão está começando. Para indicar o início de uma nova sessão, o objeto DICOM deve conter uma mensagem N-CREATE associada a essa SOP Class. Os atributos correspondentes a esta SOP Class podem ser vistos na Tabela 3.1.
- Basic Film Box SOP Class: a IOD de uma Basic Film Box engloba parâmetros através dos quais a aplicação especifica as informações relativas ao filme que será

Tabela 3.1: Tabela de atributos da Basic Film Session SOP Class para uma aplicação SCU (NEMA, 2008).

N-CREATE	
Atributo	Obrigatório
Specific Character Set	Não
Number of Copies	Não
Print Priority	Não
Medium Type	Não
Film Destination	Não
Film Session Label	Não
Memory Allocation	Não
Owner ID	Não

utilizado, como o tipo, tamanho, disposição da imagem, entre outros. O SCU deve conter no mínimo duas DIMSEs associadas a essa SOP Class: uma N-CREATE, a qual definirá a criação de uma nova Basic Film Box, e uma N-ACTION, que servirá como um comando para imprimir a última Film Box criada. Diversas Film Boxes podem estar associadas a uma mesma Film Session. Os atributos correspondentes a esta SOP Class podem ser vistos na Tabela 3.2.

- Basic Grayscale Image Box SOP Class / Basic Color Image Box SOP Class: essas SOP Classes estão associadas à anterior. Elas deverão conter as informações relativas aos píxeis da imagem (ou das imagens) que será impressa, atributos que são especificados através de uma mensagem N-SET. A aplicação não deve fazer a criação manual de uma Image Box, pois esse passo é realizado automaticamente quando é feita a criação de uma nova Film Box, a qual irá associar esses dois elementos. Os atributos correspondentes a esta SOP Class podem ser vistos na Tabela 3.3.

É possível observar que essas SOP Classes representam uma estrutura hierárquica. Ainda pode-se mencionar a Printer SOP Class, que é criada pelo próprio SCP e possibilita que a aplicação receba informações referentes ao status da impressora através de mensagens N-GET, no entanto, o padrão define esse recurso como opcional para o SCU.

Tabela 3.2: Tabela de atributos da Basic Film Box SOP Class para uma aplicação SCU (NEMA, 2008).

N-CREATE	
Atributo	Obrigatório
Image Display Format	Sim
Referenced Film Session Sequence	Sim
Referenced SOP Class UID	Sim
Referenced SOP Instance UID	Sim
Film Orientation	Não
Film Size ID	Não
Magnification Type	Não
Max Density	Não
Configuration Information	Não
Referenced Presentation LUT Sequence	Não
Referenced SOP Class UID	Não
Referenced SOP Instance UID	Não
Annotation Display Format ID	Não
Smoothing Type	Não
Border Density	Não
Empty Image Density Density	Não
Min Density	Não
Trim	Não
Illumination	Não
Reflected Ambient Light	Não
Requested Resolution ID	Não
ICC Profile	Não
N-ACTION	
Sem atributos para SCU.	

Tabela 3.3: Tabela de atributos da Basic Grayscale Image Box SOP Class para uma aplicação SCU (NEMA, 2008).

N-SET	
Atributo	Obrigatório
Image Box Position	Sim
Basic Grayscale Image Sequence	Sim
Samples Per Pixel	Sim
Photometric Interpretation	Sim
Rows	Sim
Columns	Sim
Pixel Aspect Ratio	Sim se não for 1\1
Bits Allocated	Sim
Bits Stored	Sim
High Bit	Sim
Pixel Representation	Sim
Pixel Data	Sim
Polarity	Não
Magnification Type	Não
Smoothing Type	Não
Min Density	Não
Max Density	Não
Configuration Information	Não
Requested Image Size	Não
Requested Decimate/Crop Behavior	Não
Referenced Presentation LUT Sequence	Não
Referenced SOP Class UID	Não
Referenced SOP Instance UID	Não

3.2 Implementação do recurso de impressão de imagens diagnósticas em impressoras comuns

3.2.1 Estratégia e estudo sobre o Weasis

Considerando o fato de que a impressão comum utiliza os *drivers* da própria impressora, optou-se por realizar a implementação diretamente no projeto do Weasis, tendo em vista que a linguagem Java oferece suporte a esse tipo de impressão conforme será descrito mais adiante.

Primeiramente, foi necessário estudar como o projeto Weasis está estruturado. Pelo fato da *workstation* possuir diversos recursos, seus desenvolvedores optaram por adotar uma arquitetura altamente modular (RODUI, 2010) a fim de facilitar a organização do código. Esses módulos são divididos de forma a agrupar funcionalidades semelhantes, de modo que uns são dependentes de outros conforme é possível visualizar no diagrama (Figura 3.2). Tendo em vista a necessidade de se ter fácil acesso às imagens que estão abertas para a utilização da impressão, decidiu-se adicionar o recurso ao módulo Weasis Core UI por ele já conter classes que manipulam as imagens, além da barra de ferramentas para que a funcionalidade pudesse ser adicionada como um novo ítem do menu.

3.2.2 Codificação

Após o estudo sobre o funcionamento do Weasis, a implementação do novo recurso passou a ser realizada de forma colaborativa com o desenvolvedor Nicolas Roduit, um dos principais envolvidos no projeto. Inicialmente foi implementada a impressão de uma única imagem e para isso observou-se que até o momento não havia uma maneira prática de se obter um objeto que contivesse o que estava sendo visualizado na tela. Para solucionar esse problema, foi criada a classe `ExportImage`, que contém métodos para obtenção da imagem e das informações textuais que estão sendo exibidas na tela.

Em seguida, foi criada uma janela na interface gráfica (Figura 3.3) para que o usuário pudesse selecionar preferências de impressão como o posicionamento da imagem na página e a impressão ou não de anotações (informações textuais relativas ao exame). As classes `PrintDialog` e `PrintOptions` possuem a implementação da janela gráfica e do encapsulamento das preferências respectivamente.

Com a imagem e as preferências do usuário, já se tinha as informações necessárias para serem enviadas à impressora. Para isso, foi criada a classe `ImagePrint`, a qual im-

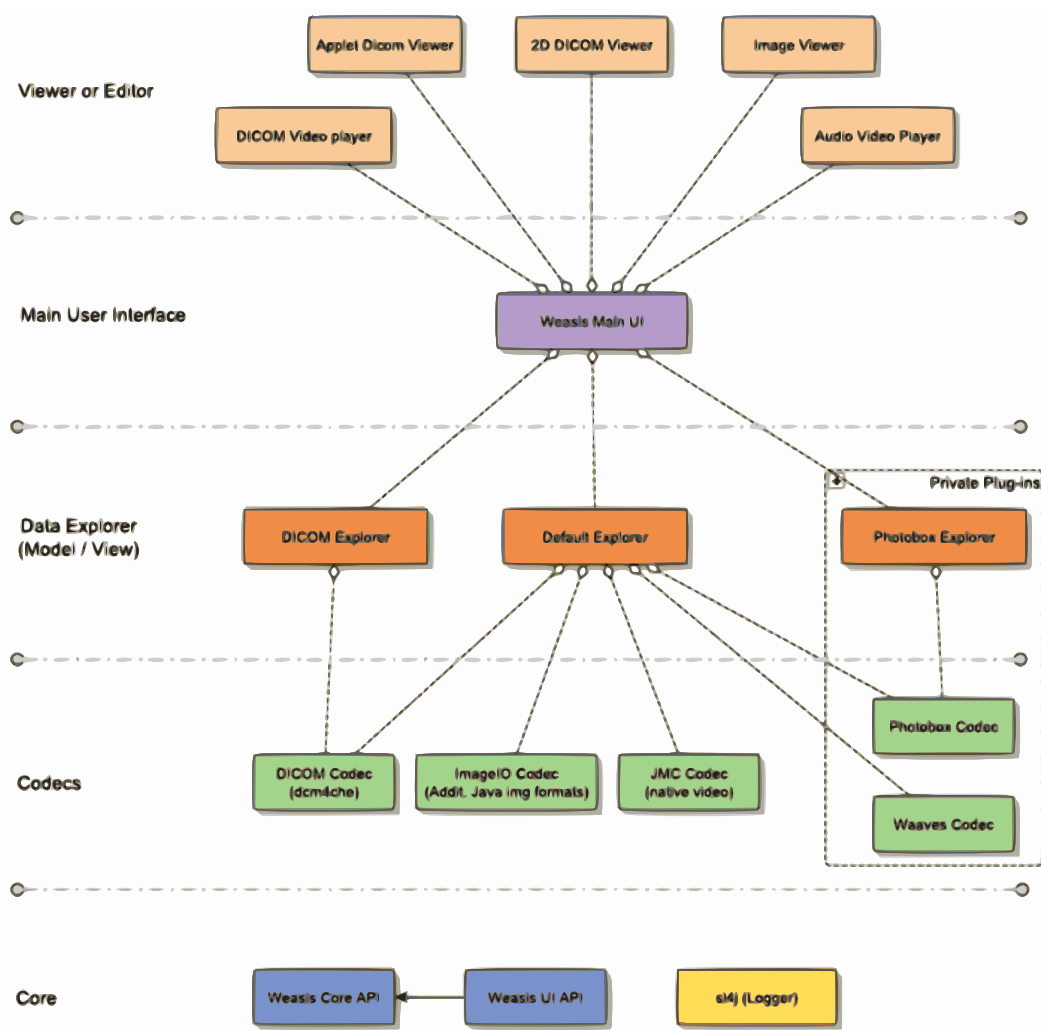


Figura 3.2: Arquitetura dos módulos do Weasis.

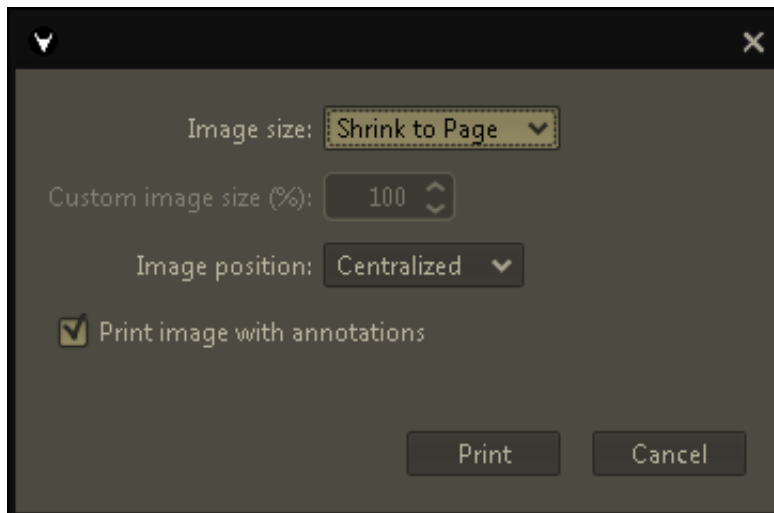


Figura 3.3: Janela da interface gráfica para impressão comum.

plementa a interface Printable do Java. Essa interface define métodos para a configuração da página através da classe PageFormat e também para a comunicação com a impressora, processo que é abstraído pela classe PrinterJob (DUBE, 2000). A aplicação cria uma nova sessão de impressão chamada de *job*, exhibe uma janela com as opções oferecidas pelo *driver* do dispositivo e envia, através do método *print()*, o comando para que a impressão inicie.

Para que fosse possível a impressão de múltiplas imagens em uma mesma folha, foi criada a classe ExportLayout. O Weasis trabalha com um sistema de layouts, de modo que é possível visualizar até 16 imagens (em uma matriz 4x4) na tela de forma simultânea. A estratégia adotada foi reproduzir na folha exatamente o que estava sendo exibido na tela, de modo que para isso foi necessário aplicar escalas para reduzir os tamanhos de imagens de acordo com o layout utilizado e com as dimensões do papel.

As classes criadas e utilizadas no processo podem ser visualizadas no diagrama (Figura 3.4).

3.3 Implementação do recurso de impressão de imagens diagnósticas em impressoras DICOM

3.3.1 Estratégia

Por se tratar de um processo mais complexo que o anterior, optou-se por realizar a implementação da impressão DICOM de maneira independente do Weasis em um momento inicial a fim de facilitar a compreensão do processo. Para isso, foi criada uma única classe

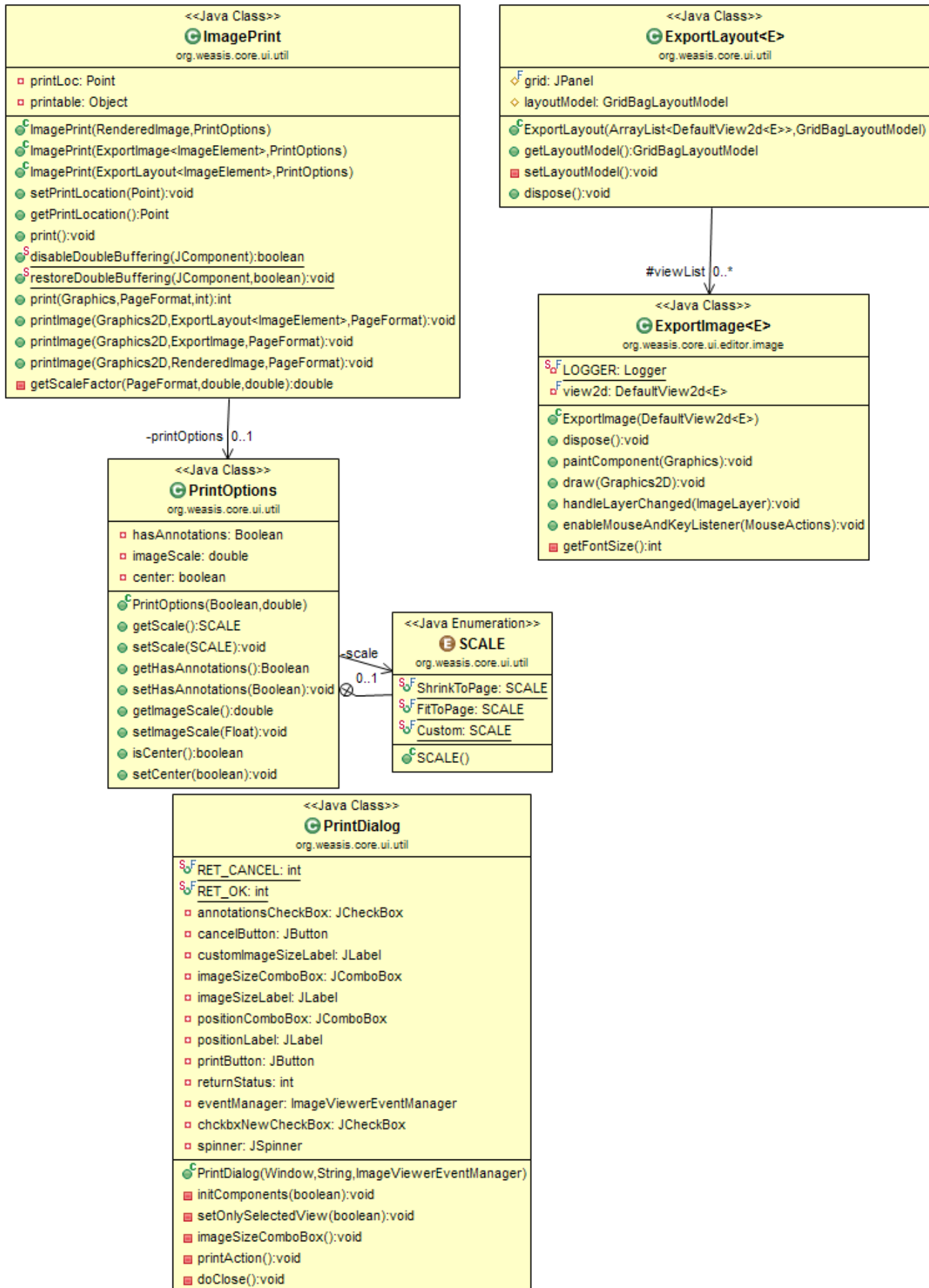


Figura 3.4: Diagrama de classes com as classes desenvolvidas e utilizadas no processo de impressão comum.

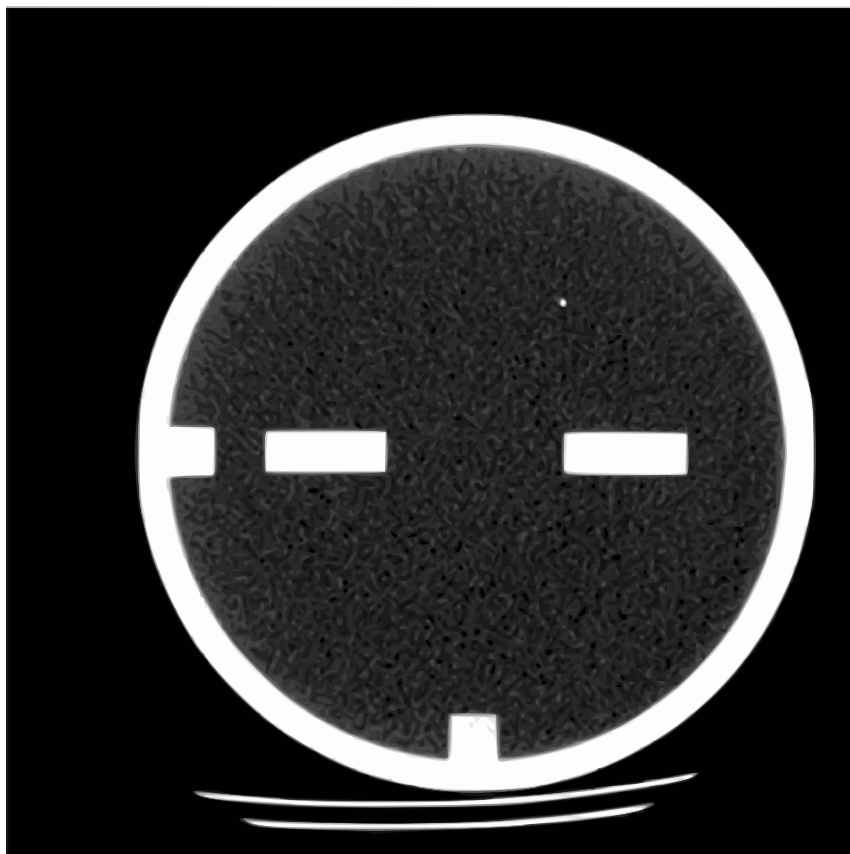


Figura 3.5: Tomografia do tipo QC utilizada na classe-teste da impressão DICOM.

para fins de teste, a qual utiliza recursos e bibliotecas do Java juntamente com o *dcm4che2 toolkit*. Essa classe possibilitou que o recurso fosse testado a partir da compilação e execução de um único arquivo que contém o mínimo necessário para se realizar a impressão de um objeto DICOM em tons de cinza (Basic Grayscale Print Management Meta SOP Class).

A imagem utilizada para teste foi uma tomografia (CT) do tipo QC (Quality Control) cujo Transfer Syntax é Implicit VR Little Endian (Figura 3.5). Imagens QC são utilizadas para fins de testes de qualidade dos tomógrafos, possibilitando que sejam avaliados aspectos como nível de ruído nas imagens geradas, contraste, resolução espacial, entre outros (HINTENLANG, 2010). Optou-se por utilizar uma imagem desse tipo por ela ser uma representação adequada de uma tomografia real, mas que não expõe dados sigilosos de médicos ou pacientes.

3.3.2 Escolha de emuladores Print SCP

Para verificar se a implementação de uma aplicação SCU para impressão DICOM estava sendo feita de maneira correta, era necessário realizar testes a cada modificação

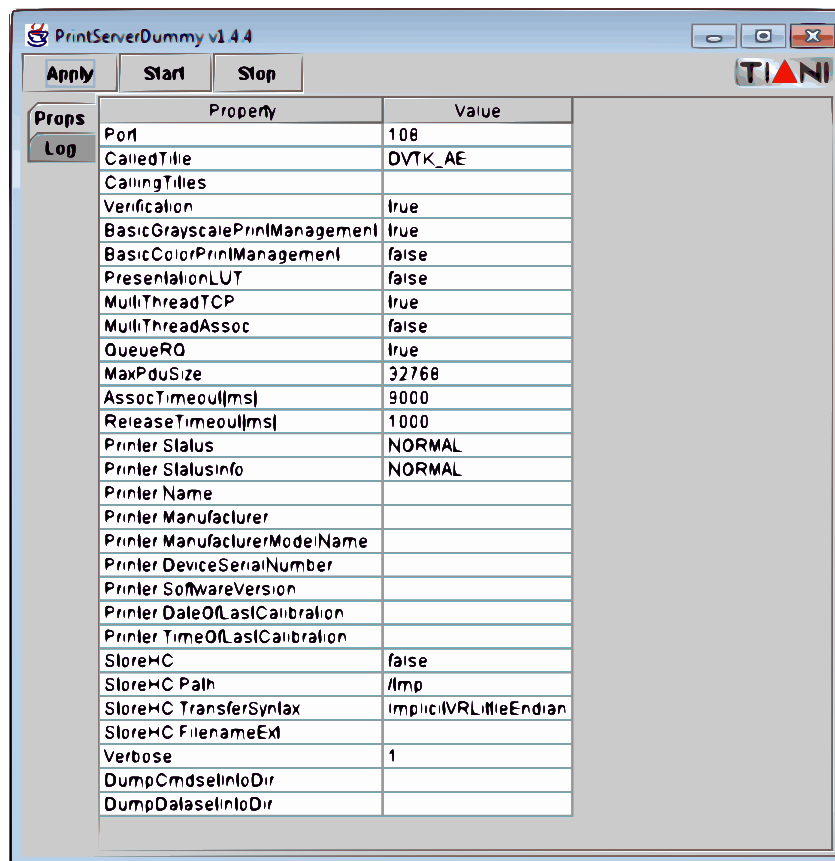


Figura 3.6: Tela inicial do PrintServerDummy para ajustes de configuração.

para ter a garantia de que ela era capaz de se comunicar com o dispositivo de impressão. No entanto, não era viável fazer todos esses testes em uma impressora DICOM real, pois se trata de um equipamento de alto custo cujo investimento não compensaria ser feito apenas para a realização de testes. Sendo assim, optou-se por adotar emuladores Print SCP durante a implementação da funcionalidade. Esses aplicativos têm como objetivo simular a comunicação entre uma aplicação SCU e uma impressora DICOM, além de informar se essa AE está em conformidade com o padrão.

Os emuladores utilizados para os testes do programa foram o PrintServerDummy (Figura 3.6) e o Dicom Validation Tool (Figura 3.7). O primeiro, apesar de oferecer facilidade na configuração, mostrou-se pouco funcional, pois somente recebe requisições através das DIMSEs e envia respostas. Em contrapartida, a segunda ferramenta, além de receber as requisições e enviar respostas, também realiza a validação do objeto DICOM, informando se todos os atributos obrigatórios estão presentes nele, e gera um *log* detalhado com essas informações.

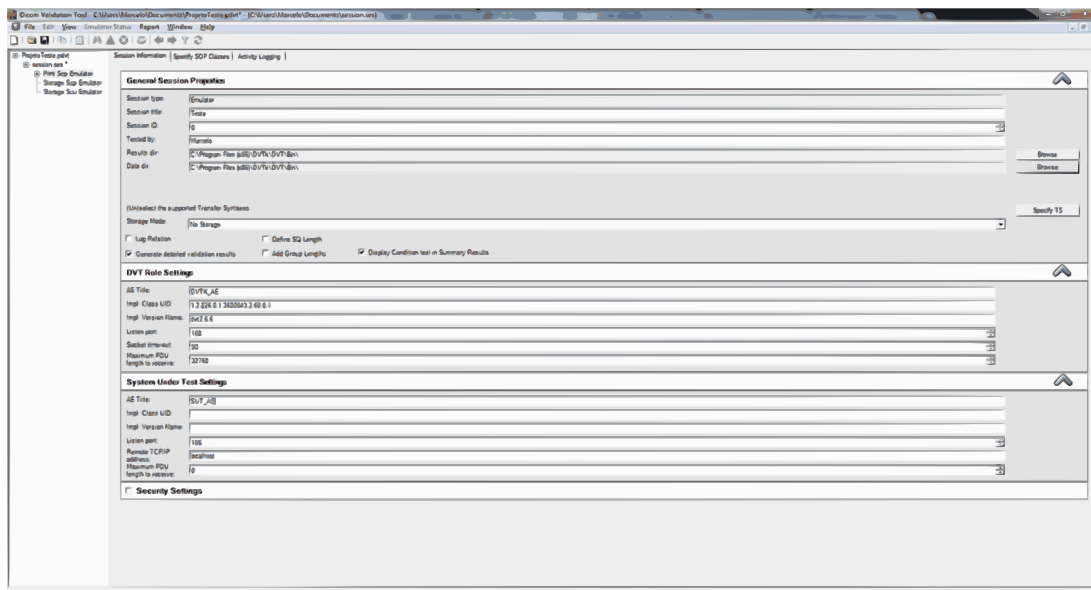


Figura 3.7: Emulador SCP do Dicom Validation Tool.

```
final File dicomFile = new File(DICOM_FILE_PATH);
final DicomInputStream dis = new DicomInputStream(dicomFile);
DicomObject dicomImage = dis.readDicomObject();
dis.close();
```

Código 3.3.1: Armazenando um arquivo DICOM em um DicomObject.

3.3.3 Codificação

Após a criação da classe para testes e da inclusão das bibliotecas do *dcm4che2 toolkit*, era necessário carregar uma imagem DICOM do disco rígido para que fosse possível acessá-la como objeto dentro do programa. A classe *DicomInputStream* do *toolkit* permite que esse processo seja feito de forma prática, pois oferece um método que, a partir de um buffer do arquivo, possibilita a conversão direta para um *DicomObject*. O método em questão é o *readDicomObject()*, cuja utilização pode ser vista no Código 3.3.1.

O passo seguinte foi conseguir estabelecer um canal de comunicação entre a aplicação e o emulador Print SCP (Código 3.3.2). O *dcm4che2* possui um pacote *net* que contém classes para a realização dessa comunicação, sendo a *NetworkConnection* e a *NetworkApplicationEntity* as duas principais nesse caso, a partir das quais é possível definir uma nova conexão e uma nova AE que irá se comunicar via rede respectivamente. Deve-se ter duas instâncias de cada uma dessas classes, uma contendo as informações da aplicação em questão (host, porta e AE title) e outra contendo esses mesmos dados relativos ao Print SCP. Além disso, é nesta parte da implementação que deve-se negociar um Presentation

```

TransferCapability[] transferCapability = {
    new TransferCapability(
        UID.BasicGrayscalePrintManagementMetaSOPClass,
        NATIVE_LE_TS, TransferCapability.SCU)
};
Device device = new Device("");
//AE local (SCU)
NetworkApplicationEntity ae = new NetworkApplicationEntity();
NetworkConnection conn = new NetworkConnection();
Executor executor = new NewThreadExecutor("SUT_AE");
//AE remota (SCP)
NetworkApplicationEntity remoteAE;
remoteAE = new NetworkApplicationEntity();
NetworkConnection remoteConn = new NetworkConnection();

conn.setPort(106);
conn.setHostname("localhost");
ae.setNetworkConnection(conn);
ae.setAssociationInitiator(true);
ae.setAETitle("SUT_AE");
ae.setTransferCapability(transferCapability);

remoteConn.setPort(108);
remoteConn.setHostname("localhost");
remoteConn.setSocketCloseDelay(0);

remoteAE.setNetworkConnection(remoteConn);
remoteAE.setAssociationAcceptor(true);
remoteAE.setAETitle("DVTK_AE");

device.setNetworkApplicationEntity(ae);
device.setNetworkConnection(conn);

```

Código 3.3.2: Definição e configuração das AEs.

Context, sendo que para isso é criada uma instância da classe *TransferCapability* (também presente no *toolkit*) e associado um Abstract Syntax UID, que nesse caso corresponde ao UID da Basic Grayscale Print Management Meta SOP Class, a um ou mais Transfer Syntax UIDs das imagens que serão impressas e o tipo de AE que está sendo desenvolvida (SCU no caso). Esse objeto criado será atribuído à instância da classe *NetworkApplicationEntity* que inicia a comunicação, ou seja, a que representa o SCU.

Por último, é necessário enviar as requisições via DIMSEs. Para isso, é criado um objeto da classe *Association*, que servirá como elo entre a AE remota e a local criadas anteriormente. Essa classe implementa todas as mensagens DIMSE presentes no DICOM em forma de métodos, sendo que para o processo de impressão foram utilizadas somente mensagens N-CREATE, N-SET e N-DELETE. Cada conjunto de atributos das SOP Classes utilizadas nas DIMSEs foram encapsulados em instâncias da classe *DicomObject*.

É possível observar no Código 3.3.3 que a mensagem N-SET relativa à Basic Grayscale Image Box SOP Class é dependente da resposta do SCP após a criação da Basic Grayscale Film Box, isso porque é necessário o UID da Image Box, que foi criada automaticamente na mensagem anterior, para alterar seus atributos.

```

DimseRSP ncreateRSP = association.ncreate(
    UID.BasicGrayscalePrintManagementMetaSOPClass,
    UID.BasicFilmBoxSOPClass,
    filmBoxUID,
    filmBoxAttrs,
    transferCapability[0].getTransferSyntax()[0]
);
ncreateRSP.next();
DicomObject imageBoxSequence = ncreateRSP.getDataset()
    .getNestedDicomObject(Tag.ReferencedImageBoxSequence);
association.nset(UID.BasicGrayscalePrintManagementMetaSOPClass,
    UID.BasicGrayscaleImageBoxSOPClass,
    imageBoxSequence.getString(Tag.ReferencedSOPInstanceUID),
    imageBoxAttrs,
    transferCapability[0].getTransferSyntax()[0]
);

```

Código 3.3.3: Resposta da mensagem N-CREATE da Film Box é armazenada para termos acesso à Image Box criada.

3.4 Integração da impressão DICOM com o Weasis

Diferentemente da impressão comum, optou-se por adicionar a impressão DICOM ao módulo DICOM Explorer do Weasis, tendo em vista que ele agrupa classes e recursos que estão diretamente relacionados com o padrão. Além disso, foi possível observar que as informações relativas às configurações da impressora deveriam ser persistentes para que não houvesse a necessidade do usuário preenchê-las a cada nova execução da *workstation*. A persistência de dados no Weasis é feita a partir do uso de arquivos de configurações no formato XML (Extensible Markup Language), sendo que o módulo DICOM Explorer já apresenta uma classe para que essas propriedades sejam lidas do arquivo (classe *Activator*), tornando necessária apenas a implementação de um interpretador para transformá-las em objetos.

Para a impressão DICOM, foi criada uma janela gráfica (Figura 3.8) baseada nos demais *softwares* analisados que possuem essa funcionalidade, de modo que o usuário pudesse definir valores para alguns atributos do processo. A interface também possibilita que impressoras DICOM sejam adicionadas ao sistema, as quais terão suas informações (descrição, *host*, porta e AE title) armazenadas no arquivo de configuração *dicomPrinters* no formato XML.

3.4.1 Adaptação do código desenvolvido inicialmente

A implementação da impressão DICOM desenvolvida na classe de testes não permitia que os valores dos atributos fossem modificados dinamicamente, i.e., em tempo de execu-

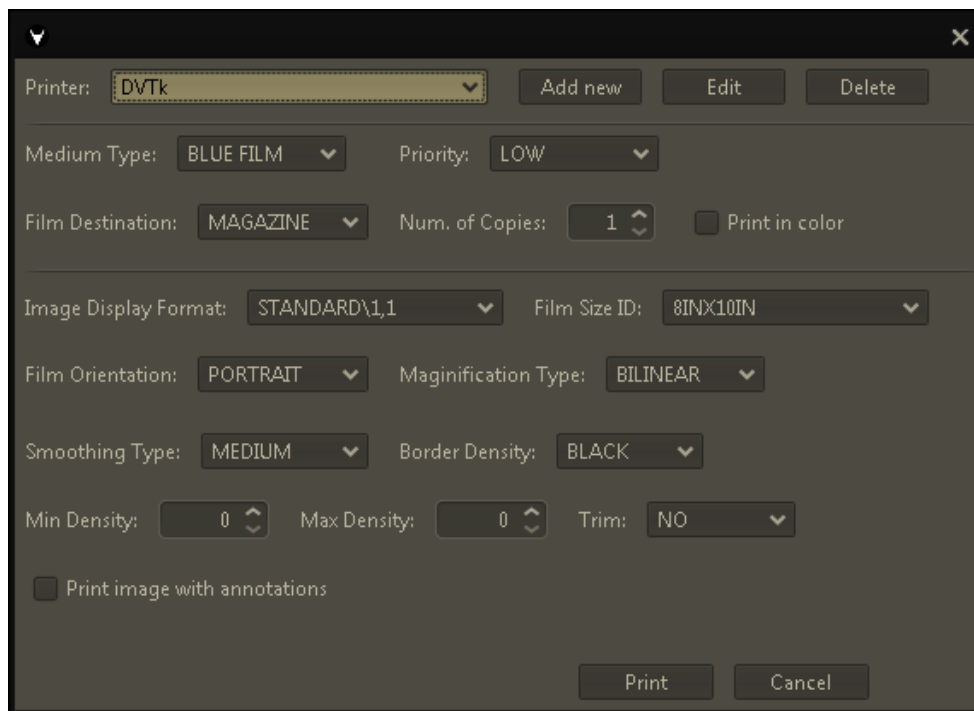


Figura 3.8: Janela da interface gráfica para impressão DICOM.

ção. Para possibilitar essas alterações no Weasis, foi criada a classe `DicomPrintOptions`, a qual armazena as propriedades de impressão definidas pelo usuário na janela da interface gráfica. Também foi criada a classe `DicomPrinter`, que contém as configurações relativas às impressoras DICOM que são recuperadas a partir do arquivo XML, interpretadas e transformadas em objetos.

Para o processo de impressão, a principal dificuldade encontrada foi a obtenção da imagem em um objeto DICOM, pois as informações dos píxeis são armazenadas em um objeto da classe `BufferedImage` do Java no momento em que uma imagem é carregada para a memória principal dentro do Weasis. Tendo em vista que os píxeis não poderiam ser enviados nesse formato através das DIMSEs, foi criado um `DicomObject` para armazenar nele essa informação, juntamente com os atributos referentes a uma `Basic Grayscale Image Sequence`, os quais podem ser visualizados na Tabela 3.3. Um detalhe a ser observado nesse processo é o fato de que a impressão DICOM em tons de cinza utiliza imagens monocromáticas, enquanto que o objeto `BufferedImage` gerado pelo Weasis possui três canais de cores (RGB), o que tornou necessária a conversão de um padrão para o outro, feita no Código 3.4.1. Após isso, o código da classe teste pôde ser adicionado ao Weasis, sendo necessário apenas substituir os valores fixos dos atributos de impressão pelos que foram definidos pelo usuário na interface gráfica.

```

private static BufferedImage convertRGBImageToMonochrome
(BufferedImage colorImage) {
    if (colorImage.getType() == BufferedImage.TYPE_BYTE_GRAY) {
        return colorImage;
    }
    BufferedImage image =
        new BufferedImage(colorImage.getWidth(),
            colorImage.getHeight(),
            BufferedImage.TYPE_BYTE_GRAY);
    Graphics g = image.getGraphics();
    g.drawImage(colorImage, 0, 0, null);
    g.dispose();
    return image;
}

```

Código 3.4.1: Método que realiza a conversão de uma imagem colorida em uma imagem monocromática no Weasis.

As classes desenvolvidas no processo podem ser visualizadas no diagrama (Fig. 3.9).

3.4.2 Adaptação do código para impressão DICOM colorida

Para possibilitar a impressão DICOM colorida, foram necessárias algumas adaptações no código. As duas principais alterações na implementação se encontram na SOP Class que gerencia o processo, conforme mencionado na seção 3.1.2, e na forma que os píxeis estão estruturados na imagem, o que resulta em modificações nos valores dos atributos da Image Box. No Código 3.4.2, é possível visualizar esses valores que são alterados de acordo com o tipo de impressão selecionado, o qual é indicado pela flag `printInColor`.

Também pôde-se observar que, para a impressão colorida, os bytes que contêm as informações dos píxeis da imagem podem ser adicionados diretamente no atributo `PixelData` do objeto DICOM, pois esses dados já estão dispostos em três canais de cores (RGB).

3.5 Testes

Inicialmente, os testes realizados em emuladores foram suficientes para verificar se a comunicação entre o Weasis e o Print SCP estava ocorrendo sem apresentar erros através do envio e recebimento de DIMSEs. No entanto, não era possível obter um resultado prático através desses testes, pois eles não geravam imagens impressas em filmes radiográficos.

Para que fosse possível testar o recurso em uma impressora real, o Instituto de Radiologia São Lucas, clínica parceira da Animati Computação Aplicada, disponibilizou o

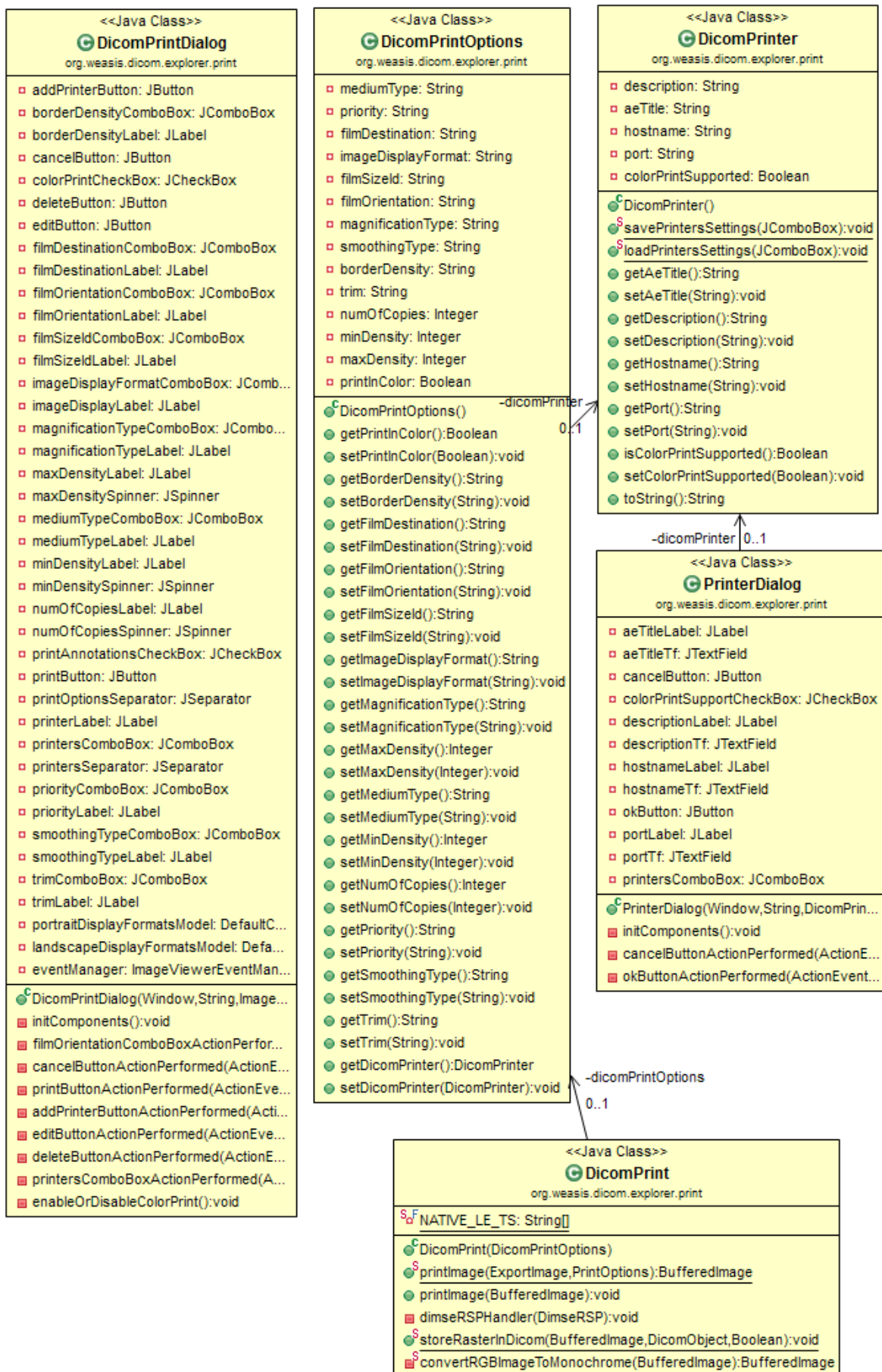


Figura 3.9: Classes criadas no Weasis referentes à impressão DICOM.

```

dcmObj.putInt(Tag.Columns, VR.US, image.getWidth());
dcmObj.putInt(Tag.Rows, VR.US, image.getHeight());
dcmObj.putInt(Tag.PixelRepresentation, VR.US, 0);
dcmObj.putString(Tag.PhotometricInterpretation, VR.CS,
    printInColor ? "RGB" : "MONOCHROME2");
dcmObj.putInt(Tag.SamplesPerPixel, VR.US, printInColor ? 3 : 1);
dcmObj.putInt(Tag.BitsAllocated, VR.US, 8);
dcmObj.putInt(Tag.BitsStored, VR.US, 8);
dcmObj.putInt(Tag.HighBit, VR.US, 7);
dcmObj.putString(Tag.TransferSyntaxUID, VR.UI,
    UID.ImplicitVRLittleEndian);
if (printInColor) {
    dcmObj.putInt(Tag.PlanarConfiguration, VR.US, 0);
} else {
    image = convertRGBImageToMonochrome(image);
}

```

Código 3.4.2: Trecho de código em que os atributos do objeto DICOM recebem valores diferentes conforme o tipo de impressão selecionado (tons de cinza ou colorida).

uso de uma impressora Kodak Carestream DryView 6850, dispositivo compatível com o padrão DICOM. Com isso, foi possível imprimir imagens de teste do tipo CT em filmes de tamanho 8x10 polegadas e verificar o funcionamento da aplicação em um contexto prático.

3.6 Resultados

Ambos os recursos de impressão implementados foram incorporados ao Weasis, possibilitando que seus usuários tenham duas alternativas para que as imagens visualizadas na tela sejam transferidas para o papel. No caso da impressão comum, é possível ter múltiplas imagens, sendo dezesseis o limite máximo, em uma mesma folha (Figura 3.10). Além disso, medições, informações textuais e alterações nas imagens também podem aparecer na versão impressa caso o usuário deseje, pois a funcionalidade foi desenvolvida seguindo a ideia WYSIWYG ("*What You See Is What You Get*").

A impressão DICOM, por sua vez, possibilita que os usuários do Weasis obtenham a imagem visualizada na tela em um filme radiográfico, oferecendo opções de configurações para que o processo seja adequado para diferentes situações e impressoras DICOM. A funcionalidade proporciona uma impressão em que o tamanho da imagem não depende do tamanho do filme (a escala é tratada pela própria impressora) e também fornece ao usuário a opção de exibir informações textuais e modificações que foram realizadas na área de visualização. A funcionalidade implementada pode ser utilizada em qualquer impressora de filmes compatível com o protocolo DICOM, sendo necessário apenas que

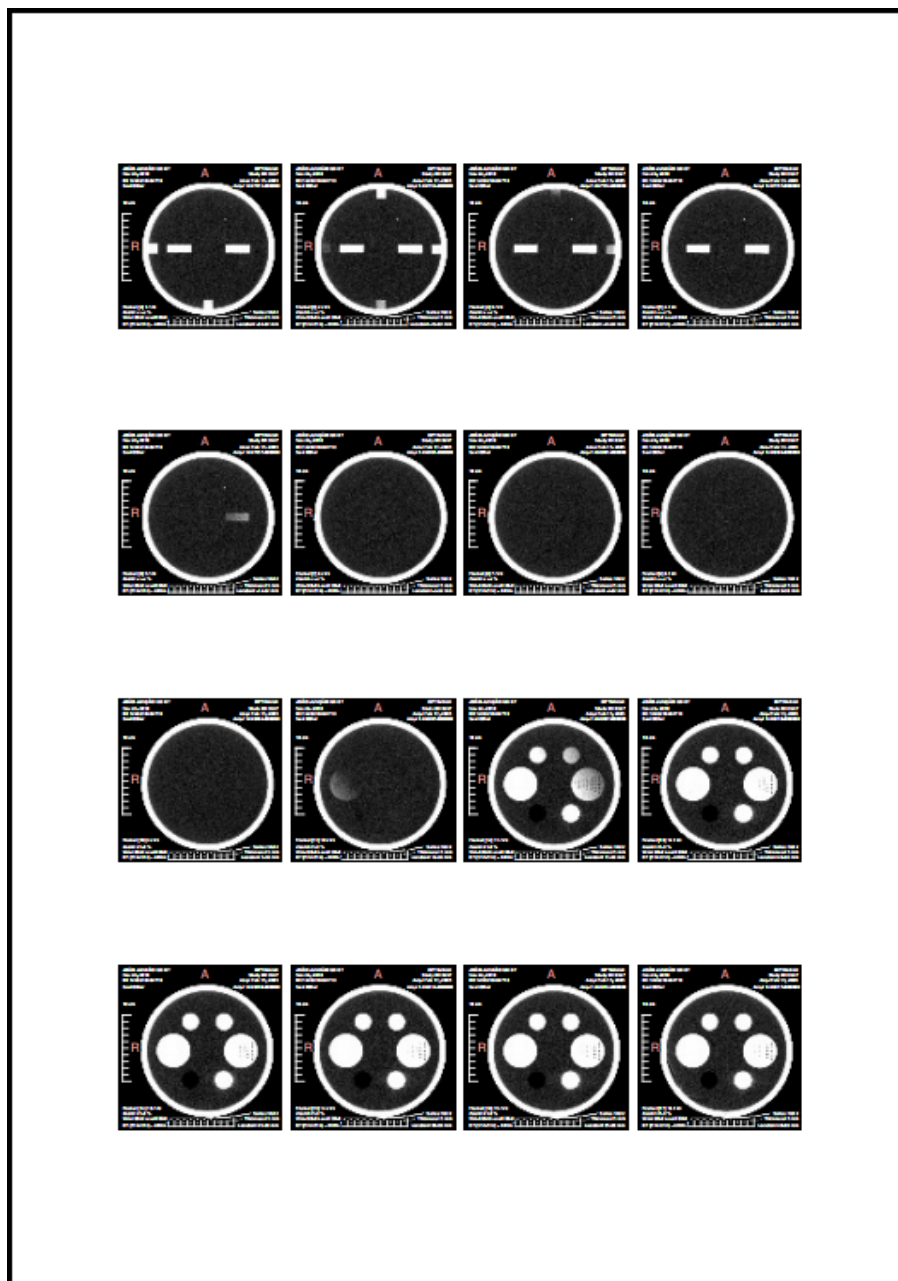


Figura 3.10: Representação da impressão comum do layout 4x4 do Weasis em uma folha A4.

o usuário configure, na aplicação, os atributos especificados no desenvolvimento deste trabalho.

4 CONCLUSÃO E TRABALHOS FUTUROS

No presente trabalho, foi implementada uma ferramenta para impressão de imagens diagnósticas. O recurso foi adicionado à *workstation* Weasis e contempla a impressão de imagens em papel e filmes radiográficos para que clínicas e profissionais da saúde possam realizar o procedimento. A ferramenta atende à demanda principal relacionada a impressão de imagens médicas, tendo em vista que apresenta recursos para que elas possam ser impressas em diferentes superfícies com as devidas alterações realizadas pelo usuário. Além disso, as opções de impressão foram disponibilizadas conforme o modelo de outros *softwares* que são referências na área, como o eFilm e o Osirix, possibilitando que novos usuários do Weasis, já adaptados a outras *workstations*, possam fazer uso do recurso de maneira intuitiva.

Para o desenvolvimento do trabalho, foram necessários estudos sobre o padrão DICOM e como ele é implementado em linguagem Java através do *dcm4che2 toolkit*, bem como sobre o Weasis e sua arquitetura modular. O projeto Weasis, até o momento da implementação deste recurso, possui pouca documentação para o desenvolvimento e integração de novos módulos, mas é possível obter suporte através do fórum de discussões na página do projeto, o que auxilia no processo de criação de *plugins* e novas funcionalidades para a *workstation*.

Como trabalho futuro, pode-se adaptar a impressão DICOM para que seja possível imprimir múltiplas imagens em um mesmo filme. O Weasis já disponibiliza uma classe para se obter todas as imagens que são exibidas nos layouts do visualizador, sendo necessário apenas alterar a forma com que os objetos DICOM são construídos no processo de impressão. Outra possibilidade que pode ser desenvolvida futuramente é a impressão sob demanda. As imagens, antes de serem impressas, são ajustadas pelo radiologista responsável, no entanto, essas modificações não podem ser salvas, fazendo com que haja a

necessidade de se enviar as imagens para a impressora logo após alterações como uma mudança nos valores de brilho/contraste, por exemplo. Para solucionar esse problema, o DICOM oferece uma estrutura denominada Presentation State, a qual armazena somente as opções de visualização associadas a uma determinada imagem, possibilitando que ela seja impressa a qualquer momento já com as devidas alterações previamente realizadas.

REFERÊNCIAS

ANIMATI. **Animati PACS**. Disponível em: <http://animati.com.br/servicos/saude/animati-pacs/>.

BLOCKER, J. **A History of Digital Radiography**. Disponível em: <http://ezinearticles.com/?A-History-of-Digital-Radiography&id=4600271>.

DICOM Print Service - Medical Connections. Disponível em: http://www.medicalconnections.co.uk/kb/DICOM_Print_Service.

DUBE, J. **Printing in Java**. Disponível em: <http://www.javaworld.com/javaworld/jw-10-2000/jw-1020-print.html>.

EVANS, D. **A Very Basic DICOM Introduction**. Disponível em: <http://www.dcm4che.org/confluence/display/d2/A+Very+Basic+DICOM+Introduction>.

EVANS, D. et al. **Who uses dcm4che?** Disponível em: <http://www.dcm4che.org/confluence/pages/viewpage.action?pageId=393>.

GAHLEITNER, A. et al. Dry versus Conventional Laser Imagers: film properties and image quality. **Radiology**, [S.l.], v.210, p.871–875, 1999.

HINTENLANG, D. **CT Scanner QC**. Disponível em: http://www.acmp.org/meetings/scottsdale_2004/Hintenlang.ppt.

HOWARD, D. **What is DICOM Data?** Disponível em: http://www.ehow.com/info_12200680_dicom-data.html.

KÖRNER, M. et al. Advances in Digital Radiography: physical principles and system overview. **RadioGraphics**, [S.l.], v.27, p.675–686, 2007.

NEMA. **DICOM Standard**. Disponível em: <http://medical.nema.org/standard.html>.

PIANYKH, O. S. **Digital Imaging and Communications in Medicine (DICOM) - A Practical Introduction and Survival Guide**. 1.ed. [S.l.]: Springer, 2008.

RODUI, N. **Weasis**. Disponível em: <http://www.dcm4che.org/confluence/display/WEA/Home>.