

Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA  
Engenharia de Software

# **Proposta de Solução para Geração de Mídias de Instalação Semiautomatizada de Sistemas Operacionais Linux, Baseado nas Melhores Abordagens da Comunidade de Software Livre**

Autor: Wilton da Silva Rodrigues  
Orientador: Prof. Dr. Wander C. M. Pereira da Silva

Brasília, DF  
2018



Wilton da Silva Rodrigues

**Proposta de Solução para Geração de Mídias de  
Instalação Semiautomatizada de Sistemas Operacionais  
Linux, Baseado nas Melhores Abordagens da  
Comunidade de Software Livre**

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. Wander C. M. Pereira da Silva

Brasília, DF

2018

---

Wilton da Silva Rodrigues

Proposta de Solução para Geração de Mídias de Instalação Semiautomatizada de Sistemas Operacionais Linux, Baseado nas Melhores Abordagens da Comunidade de Software Livre/ Wilton da Silva Rodrigues. – Brasília, DF, 2018-

34 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Wander C. M. Pereira da Silva

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA , 2018.

1. sistema operacional. 2. automatizado. I. Prof. Dr. Wander C. M. Pereira da Silva. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Proposta de Solução para Geração de Mídias de Instalação Semiautomatizada de Sistemas Operacionais Linux, Baseado nas Melhores Abordagens da Comunidade de Software Livre

CDU 02:141:005.6

---

Wilton da Silva Rodrigues

**Proposta de Solução para Geração de Mídias de  
Instalação Semiautomatizada de Sistemas Operacionais  
Linux, Baseado nas Melhores Abordagens da  
Comunidade de Software Livre**

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

---

**Prof. Dr. Wander C. M. Pereira da  
Silva**  
Orientador

---

**Prof. Dr. André Luiz Peron Martins  
Lanna**  
Convidado 1

---

**Prof. M.Sc. Ricardo Ajax Dias  
Kosloski**  
Convidado 2

Brasília, DF  
2018

# Resumo

Diante da complexidade enfrentada pelas grandes organizações para realizar o processo de instalação de Sistemas Operacionais customizados em grandes quantidades de computadores a necessidade por automatização deste processo já se tornou uma realidade no cotidiano destas organizações que buscam por meios de maximizar o desempenho dos seus colaboradores durante o processo de instalação, além de minimizar o tempo gasto nesta tarefa. Desta forma, a adoção de mídias de instalação personalizadas tem sido amplamente utilizada, através da criação de um *live-CD* responsável pela automatização do processo exigindo interação mínima do usuário. Assim foi realizado um levantamento das melhores práticas e ferramentas utilizadas pelas comunidades de *Software Livre* na criação de mídias de instalação de sistemas operacionais *Unix Like*.

**Palavras-chaves:** Sistema Operacional. Automatizado. *Live-CD*. Mídia de Instalação.

# Abstract

Faced with the complexity faced by large organizations to carry out the process of installing customized Operating Systems in large quantities of computers, the need for automation of this process has already become a reality in the daily life of these organizations that seek by means of maximizing the performance of their collaborators during the installation process, in addition to minimizing the time spent on this task. In this way, the adoption of customized installation media has been widely used, through the creation of a live CD responsible for the automation of the process, requiring minimal user interaction. Thus, a survey was made of the best practices and tools used by the Free Software communities in the creation of installation media for Unix Like operating systems.

**Key-words:** Operating System. Automated. Live-CD. Installation Media.

# Lista de ilustrações

Figura 1 – Diagrama de Classes do Instalador Java. . . . .	30
Figura 2 – Tela de Informações do Instalador Java. . . . .	31
Figura 3 – Tela de Progresso do Instalador Java. . . . .	32

# Lista de tabelas

Tabela 1 – Características das Distribuições. . . . .	19
---	----



# Lista de Códigos Fonte

2.1	Gerar um <i>live-CD</i> Arch com a estrutura baseline . . . . .	20
2.2	Gerar um <i>live-CD</i> OpenBSD com a estrutura siteXX.tgz . . . . .	20
2.3	Gerar um <i>live-CD</i> Debian com a versão Jessie . . . . .	21
4.1	Script para organizar a estrutura . . . . .	27
4.2	Ferramenta automatizada para Gerar um <i>live-CD</i> . . . . .	29

# Lista de abreviaturas e siglas

SO	Sistema Operacional
QP	Questão de Pesquisa
RSL	Revisão Sistemática de Literatura
ISO	Organização Internacional para Padronização
RFC	Pedido de Comentários

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>12</b>
1.1	Contextualização do tema	12
1.2	Justificativa	13
1.3	Objetivos	13
1.4	Questões de Pesquisa	14
1.5	Estrutura do trabalho	14
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>15</b>
2.1	Tipos de <i>Live-CD</i>	15
2.1.1	<i>Live-CD</i> de Manutenção	15
2.1.2	<i>Live-CD</i> de Instalação	16
2.2	Filosofias de <i>Live-CD</i>	17
2.2.1	Arch Linux	17
2.2.2	OpenBSD	18
2.2.3	Debian	18
2.3	Soluções Existentes	19
2.3.1	Archiso	20
2.3.2	Mkhybrid	20
2.3.3	Simple-CDD	21
2.4	Semelhanças	22
2.4.1	syslinux	22
2.4.2	isolinux	22
2.4.3	chroot	22
2.4.4	mksquashfs	23
2.4.5	xorriso	23
<b>3</b>	<b>METODOLOGIA</b>	<b>24</b>
3.1	Revisão Sistemática de literatura	24
3.2	Pesquisa Bibliográfica	25
3.3	Escolha de Abordagem	26
<b>4</b>	<b>RESULTADOS</b>	<b>27</b>
4.1	Ferramenta GeralSO	27
4.2	Instalador Java	29
4.3	Homologação do Instalador	30

<b>5</b>	<b>CONCLUSÕES</b> . . . . .	<b>31</b>
<b>5.1</b>	<b>Instância de Solução</b> . . . . .	<b>31</b>
<b>5.2</b>	<b>Possíveis aplicações</b> . . . . .	<b>32</b>
<b>5.3</b>	<b>Trabalhos futuros</b> . . . . .	<b>32</b>
	 <b>REFERÊNCIAS</b> . . . . .	 <b>33</b>

# 1 Introdução

## 1.1 Contextualização do tema

Os sistemas operacionais se tornaram softwares indispensáveis para a vida cotidiana. Um sistema operacional é, segundo (SILBERSCHATZ; GALVIN, 2014), um programa que gerencia o *hardware* de um computador além de fornecer uma plataforma para a execução de programas aplicativos e de atuar como intermediário entre o usuário e o hardware do computador.

Devido a esse papel tão importante, durante a evolução dos sistemas operacionais a necessidade de tornar esses sistemas cada vez mais confiáveis e seguros foi aumentando de forma gradativa. Nesse sentido, uma das principais preocupações é que os recursos do computador sejam protegidos contra acessos não autorizados, destruição ou alteração maliciosa e introdução acidental de inconsistências (SILBERSCHATZ; GALVIN, 2014).

Além disso, há ainda um ponto muito importante a respeito dos Sistemas Operacionais (SO): o processo de instalação, que é o passo intermediário entre o *download* da mídia de instalação do SO e a utilização do mesmo no *hardware* do computador.

Os processos de instalação podem ser divididos, segundo (AUTHORITY, 2006), em três métodos principais:

Instalação assistida, neste método usado principalmente por usuários domésticos durante a instalação, o usuário permanece no computador para fornecer todas as informações solicitadas durante o processo de instalação.

Instalação por rede, com este método, os arquivos de instalação do SO estão localizados em um compartilhamento de rede, portanto, nenhuma mídia de instalação é necessária. É frequentemente usado em empresas de pequeno porte.

Instalação automatizada, neste método não há a exigência de que o usuário forneça informações durante a configuração, pois as informações necessárias são armazenadas em um arquivo. Isso reduz ou até mesmo elimina a intervenção do usuário durante o processo de instalação. A instalação automatizada é normalmente usada por empresas de grande porte, já que proporciona uma economia significativa de tempo ao instalar várias cópias.

No que se refere aos Softwares Livres, existem diversas abordagens, adotadas pela comunidade quanto aos processos de instalação dos Sistemas *Unix Like*<sup>1</sup>. Desde processos de instalação totalmente automatizados e com interfaces gráficas amigáveis para o usuário,

---

<sup>1</sup> Se refere aos sistemas operacionais que compartilham muitas das características do original UNIX, que foi escrito em 1969 por Ken Thompson na Bell Labs, e seu sucessores.

como nos casos do Debian<sup>2</sup> e do Ubuntu<sup>3</sup> quanto processos semiautomatizados através de *scripts* de instalação, como no caso do OpenBSD<sup>4</sup>. Além de processos totalmente manuais, onde o usuário é responsável por executar todas as ações para que seja feita a instalação com sucesso, como no caso do ArchLinux<sup>5</sup>.

## 1.2 Justificativa

A necessidade de instalação de Sistemas Operacionais personalizados com configurações muito semelhantes já é uma realidade nos dias de hoje, seja em *desktops* de bibliotecas ou laboratórios, *totems*, ou terminais de autoatendimento. Todos estes equipamentos terão características, configurações e funcionamento muito parecidos.

Neste cenário, o processo de instalação assistida se torna muito custoso e demorado. Visto que se houver um montante de 1000 máquinas para serem instaladas será necessária a intervenção manual e espera de todo o tempo de instalação para cada uma destas máquinas. Além de todo o processo de customização ao final da instalação.

Por conta disto, a geração de uma mídia de instalação que reduza estes problemas se torna uma solução bastante atrativa para organizações que lidam com grandes quantidade de requisições de instalação.

O uso de um *live-CD* também resolve o problema de instalação em equipamentos que se encontram espalhados geograficamente e reduzem os custos de manter uma infraestrutura de servidores centrais para serem acessados através do processo de instalação por rede.

## 1.3 Objetivos

O objetivo geral do presente trabalho consiste em uma Proposta de Solução para Geração de Mídias de Instalação Semiautomatizada de Sistemas Operacionais Linux, Baseado nas Melhores Abordagens da Comunidade de Software Livre.

Como objetivos específicos foram definidos os seguintes itens:

- Detalhar as principais soluções existentes.
- Apontar as principais diferenças entre as soluções levantadas.
- Definir uma solução para ser executada como caso prático no trabalho.

---

<sup>2</sup> <[www.debian.org](http://www.debian.org)>

<sup>3</sup> <[www.ubuntu.com](http://www.ubuntu.com)>

<sup>4</sup> <[www.openbsd.org](http://www.openbsd.org)>

<sup>5</sup> <[www.archlinux.org](http://www.archlinux.org)>

## 1.4 Questões de Pesquisa

De acordo com as considerações acima, foram formuladas as seguintes questões de pesquisa que orientarão a condução do presente trabalho:

**QP1:** Como gerar um *live-CD* de instalação de um Sistema Operacional Linux?

**QP2:** Qual a melhor abordagem para a geração de um *live-CD* de instalação de Sistema Operacional Linux?

## 1.5 Estrutura do trabalho

No Capítulo 2, serão tratadas as soluções existentes no que diz respeito à criação de imagens de instalação de SO Linux, além das ferramentas utilizadas nestas soluções, de forma que ao final haja insumos suficientes para que seja possível responder à **QP1**.

Em seguida, no Capítulo 3, será apresentada a estrutura metodológica utilizada para atingir os objetivos propostos e, conseqüentemente, responder às questões de pesquisa levantadas. Também será decidido, de acordo com a fundamentação levantada, qual abordagem será seguida na parte prática do presente trabalho.

Já no Capítulo 4, será apresentada a solução obtida, as ferramentas escolhidas para atingir esta solução e os procedimentos necessários. Desta forma, respondendo à **QP2**.

Por fim, no Capítulo 5, serão retomados os objetivos do trabalho e a apresentação de possíveis aplicação da solução e trabalhos futuros.

## 2 Fundamentação Teórica

Segundo (UTUNI et al., 2005) um *live-CD* Linux é uma distribuição Linux portátil executada a partir de um CD inicializável que contém uma coleção de aplicativos, geralmente gratuitos (*free software*), que são *softwares* que dão ao usuário a liberdade de compartilhar, estudar e modificá-lo (FOUNDATION, 2018). Embora, atualmente não seja mais limitado apenas aos CD's, mas para qualquer mídia de instalação. Visto que é possível ter o mesmo resultado utilizando também dispositivos de mídia removível como *pen drive*, cartões de memória *flash* (*Micro-SD*, *SD Card*, etc.) ou qualquer outro dispositivo similar, como afirma (CROWLEY, 2009).

Por *design*, um *live-CD* Linux, a princípio, não impacta o *hardware* do sistema onde o mesmo é iniciado (CROWLEY, 2009), visto que o mesmo é completamente autônomo, e a partir dele é possível executar qualquer sistema operacional contido nele, tudo sem fazer qualquer alteração no equipamento, exigindo apenas o suporte básico de hardware: teclado, monitor, mouse e rede. Por isso existe à ampla adoção deste tipo de sistema para construção de mídias de manutenção, como listado por (OUTIN, 2017), que traz mais de 300 exemplos diferentes de mídias com este objetivo.

O *live-CD* só existe por si só após ser gravado em alguma mídia removível, no presente trabalho quando for referido o processo de gerar o arquivo com extensão *.iso* para ser gravado na mídia, o mesmo será tratado como imagem de instalação.

Embora um *live-CD* possa ser utilizado apenas como mídia de manutenção, ele também é largamente utilizado como mídia de instalação. Ou seja, é o elemento responsável por organizar logicamente toda a estrutura física onde o Sistema Operacional irá ser instalado, como será detalhado a seguir.

### 2.1 Tipos de *Live-CD*

Como já mencionado anteriormente, existem dois grandes grupos em que é possível dividir os tipos de *live-CD*. O primeiro é o *Live-CD* de Manutenção e o segundo *Live-CD* de Instalação.

#### 2.1.1 *Live-CD* de Manutenção

Os *Live-CDs* de Manutenção têm como principal objetivo, restaurar um estado anterior de funcionamento, alterar o particionamento de discos, recuperação de dados ou ainda remoção de *malwares*.



Geralmente, este tipo de *live-CD* vem com uma variedade de ferramentas técnicas úteis, como diagnósticos, análise forense, *benchmarking*, antivírus, recuperação de disco, ferramentas de segurança além de várias outras. Estas ferramentas, em sua maioria *softwares livres* (FOUNDATION, 2018), têm como objetivo ajudar o usuário a concluir com sucesso as tarefas críticas que precisam ser executadas para atingir o objetivo de reestabelecer o funcionamento do sistema.

Existe uma série de *live-CDs* desta categoria, a seguir serão listados os principais:

1. **System Rescue CD<sup>6</sup>**: O SystemRescueCd é um disco de recuperação do sistema Linux disponível como um CD-ROM inicializável ou dispositivo USB para administrar ou reparar sistemas e dados após uma falha. Ele vem com vários utilitários do sistema Linux, como GParted, fsarchiver, ferramentas de sistema de arquivos e outras ferramentas básicas (SYSTEMRESCUE, 2006).
2. **Clonezilla<sup>7</sup>**: O Clonezilla Live é uma pequena distribuição GNU/Linux inicializável usada para implantar vários computadores simultaneamente (CLONEZILLA, 2004).
3. **Gparted Live<sup>8</sup>**: O GParted Live é uma pequena distribuição GNU/Linux inicializável que permite você usar todos os recursos das versões mais recentes do aplicativo de particionamento de discos GParted (GPARTED, 2004).

O presente trabalho se concentrou na categoria de *live-CD* que será detalhada em Subseção à posteriori.

### 2.1.2 *Live-CD* de Instalação

Os *Live-CDs* de Instalação têm como principal objetivo, permitir o processo de instalação do Sistema Operacional sobre o disco rígido do equipamento. Além disso, esta categoria de *live-CD* também permite testar as funcionalidades do Sistema Operacional sem a necessidade de efetivamente ter que realizar este processo de instalação.

Embora haja um grande volume de *live-CDs* deste tipo, por questões de cronograma e escopo, no presente trabalho iremos abordar apenas três distribuições Linux:

1. **Arch Linux<sup>9</sup>**: O Arch Linux é uma distribuição GNU/Linux de uso geral desenvolvida de forma independente e versátil o suficiente para atender a qualquer função. O desenvolvimento se concentra na simplicidade, no minimalismo e na elegância do código. O Arch é instalado como um sistema básico mínimo, configurado pelo

<sup>6</sup> <<http://www.system-rescue-cd.org/>>

<sup>7</sup> <<https://clonezilla.org/clonezilla-live.php>>

<sup>8</sup> <<https://gparted.org/livecd.php>>

<sup>9</sup> <<https://www.archlinux.org>>

usuário no qual seu próprio ambiente ideal é montado, instalando apenas o que é necessário ou desejado para seus propósitos exclusivos. Os utilitários de configuração da interface gráfica não são fornecidos oficialmente ([ARCHLINUX.ORG](http://ARCHLINUX.ORG), 2001b).

2. **OpenBSD<sup>10</sup>**: O projeto OpenBSD produz um sistema operacional semelhante ao UNIX, baseado em multi-plataforma. Nossos esforços enfatizam portabilidade, padronização, correção, segurança proativa e criptografia integrada. Como um exemplo do efeito que o OpenBSD tem, o popular software OpenSSH vem do OpenBSD ([OPENBSD.ORG](http://OPENBSD.ORG), 1997b).
3. **Debian<sup>11</sup>**: O Debian é um sistema operacional livre que é acompanhado de um conjunto de programas básicos e utilitários que fazem o computador funcionar. O Debian possui mais de 51.000 pacotes contendo softwares pré-compilados e distribuídos em um formato que torna fácil a instalação deles na máquina ([DEBIAN.ORG](http://DEBIAN.ORG), 1993).

A escolha destas três distribuições se deveu ao fato de que cada uma delas é adepta de uma filosofia de instalação em particular. Estas filosofias serão abordadas com detalhes na Seções seguintes. E, em seguida, também será apresentada a forma como cada uma destas distribuições gera seu *live-CD* .

## 2.2 Filosofias de *Live-CD*

Nesta Seção, serão apresentadas as filosofias de desenvolvimento de cada uma das distribuições escolhidas. Detalhar a filosofia de cada uma destas distribuições é importante, pois além desta filosofia estar presente na forma como o Sistema Operacional é construído, ela também está presente em como o *live-CD* de instalação da distribuição é gerado e em como se dá o processo de instalação do mesmo.

### 2.2.1 Arch Linux

A filosofia de desenvolvimento do Arch Linux, segundo ([ARCHLINUX.ORG](http://ARCHLINUX.ORG), 2001b), se baseia em cinco princípios:

1. **Simplicidade**: O Arch Linux define simplicidade sem adições ou modificações desnecessárias. Ele envia o software conforme liberado pelos desenvolvedores originais.
2. **Modernidade**: O Arch Linux se esforça para manter as versões mais recentes do seu software, desde que a quebra do sistema possa ser razoavelmente evitada. Ele

---

<sup>10</sup> <<https://www.openbsd.org>>

<sup>11</sup> <<https://www.debian.org>>

é baseado em um sistema de liberação contínua, que permite uma instalação única com upgrades contínuos.

3. **Pragmatismo:** Arch é uma distribuição pragmática e não ideológica. Os princípios aqui são apenas diretrizes úteis. Em última análise, as decisões de design são tomadas caso a caso por consenso entre os desenvolvedores. A análise técnica e o debate baseados em evidências são o que importa, não a política ou a opinião popular.
4. **Centrado no Usuário:** Enquanto muitas distribuições GNU/Linux tentam ser mais amigáveis ao usuário, o Arch Linux sempre foi e sempre permanecerá centrado no usuário. A distribuição destina-se a atender às necessidades daqueles que contribuem para isso, em vez de tentar atrair o maior número de usuários possível. Ele é direcionado ao usuário proficiente do GNU/Linux, ou a qualquer pessoa com uma atitude do tipo "faça você mesmo" que esteja disposta a ler a documentação e resolver seus próprios problemas.
5. **Versatilidade:** O Arch Linux é uma distribuição de propósito geral. Após a instalação, apenas um ambiente de linha de comando é fornecido: em vez de levar diversos pacotes desnecessários, o usuário tem a capacidade de criar um sistema personalizado.

## 2.2.2 OpenBSD

Em seguida, temos a filosofia do OpenBSD, segundo ([OPENBSD.ORG](https://openbsd.org), 1997b), os dois princípios a seguir são os mais importantes na filosofia do SO:

1. **Segurança:** O OpenBSD trabalha duro para manter a política de licença e segurança do projeto. Por esse motivo, algumas versões mais recentes de alguns softwares que não atendem às metas de licença ou segurança do projeto não foram e nunca poderão ser integradas ao OpenBSD. A segurança e o licenciamento gratuito nunca ficarão em segundo plano para ter o maior número de versão.
2. **Experiência do Usuário:** Novos usuários para o OpenBSD devem olhar para sua própria experiência: se seu único conhecimento de Unix é alguma experiência com uma variante do Linux, você pode achar o OpenBSD "estranho".

## 2.2.3 Debian

Já a filosofia do Debian, segundo ([DEBIAN.ORG](https://www.debian.org), 1993), embora guiada também por outros princípios, tem como principais os três a seguir:

1. **Facilidade de uso:** O Debian preza por ser o mais amigável possível ao usuário. Qualquer usuário com conhecimento mínimo consegue realizar o processo de instalação do SO e utilizar o mesmo.
2. **Segurança e Confiabilidade:** O Debian foca na estabilidade e desempenho do sistema. A liberação de pacotes apenas ocorre após vários testes de integração e testes da comunidade antes que o mesmo seja liberado oficialmente nos repositórios.
3. **Disponibilidade de Software:** O Debian disponibiliza repositórios com diversos programas pré compilados para a arquitetura e versão do SO de forma que com apenas um comando é possível instalar e já estar apto a utilizar o programa, sem necessidade de compilação ou processos de instalação complexos.

A Tabela 1 traz de forma resumida as principais características das distribuições, baseado nas filosofias de cada uma.

Característica	Debian	OpenBSD	Arch Linux
Facilidade de Instalação	Alta	Média	Baixa
Facilidade de Uso	Alta	Média	Baixa
Experiência do Usuário	Baixa	Média	Alta
Atualização de Pacotes	Média	Baixa	Alta
Pronto pra usar	Sim	Não	Não

Tabela 1 – Características das Distribuições.

Também é possível notar que as distribuições que possuem uma filosofia mais centrada na experiência do usuário, tendem a criar processos de instalações mais manuais e que demandam alto grau de conhecimento técnico.

No caso das distribuições que focam menos na experiência do usuário e mais em outros quesitos, como segurança no caso apresentado neste trabalho, tendem a oferecer processos de instalação semiautomatizados, onde o usuário ainda precisa de um certo grau de conhecimento, porém o a instalação o auxilia e automatiza várias etapas do processo.

Por fim, no caso das distribuições que focam em ser mais amigáveis ao usuário, obtemos uma instalação facilitada e totalmente automatizada onde até mesmo os usuários mais inexperientes conseguem realizar com sucesso o processo.

## 2.3 Soluções Existentes

Existem diversas soluções referentes à geração automatizada de imagens de instalação que são disponibilizadas pelos distribuidores responsáveis pelo desenvolvimento do Sistema Operacional. Na maioria das vezes, os próprios desenvolvedores utilizam estas ferramentas para a distribuição oficial das suas imagens de instalação.

### 2.3.1 Archiso

O Archiso é a ferramenta oficial de geração de imagens para o Arch Linux, apresentado na Subseção 2.1.2. É composta por um pequeno conjunto de scripts *bash* capazes de construir imagens *live-CDs* totalmente funcionais do Arch Linux. Além de ser a ferramenta usada para gerar as imagens oficiais, por ser uma ferramenta muito genérica, pode ser usada para gerar qualquer coisa desde sistemas de resgate, discos de instalação, até outros *live-CDs* (ARCHLINUX.ORG, 2001a).

O Archiso consiste basicamente de uma série de scripts e configurações pré definidas no qual o principal é o *mkarchiso*. De forma que com os comandos apresentados na listagem 2.1 são o suficiente para criar uma imagem de instalação baseada no perfil *baseline* pré configurado.

```
1 #!/bin/bash
2
3 # Em um sistema Arch Linux
4
5 pacman -S archiso
6
7 mkdir -p archive/{airootfs,out}
8
9 cp -r /usr/share/archiso/configs/baseline/ archive
10
11 /usr/bin/mkarchiso
```

Código Fonte 2.1 – Gerar um *live-CD* Arch com a estrutura *baseline*

### 2.3.2 Mkhybrid

O mkhybrid é uma das ferramentas utilizadas para gerar imagens de instalação do OpenBSD. Embora não seja a única, ela será abordada neste trabalho como a representante desta distribuição. Os scripts de instalação e atualização do OpenBSD permitem a seleção de um conjunto criado pelo usuário chamado *siteXX.tgz*, onde *XX* é a versão de lançamento. Ele é instalado por último, portanto, pode ser usado para complementar e modificar os arquivos de uma instalação padrão (OPENBSD.ORG, 1997a).

```
1 #!/bin/bash
2
3 # Em um sistema OpenBSD
4
5 mkdir openbsd
6 mkdir SITEXX
7
8 # Adicione todos as customizacoes necessarias dentro da pasta SITEXX
9
```

```
10 tar cvzf siteXX.tgz -C FILES SITEXX
11
12 cp siteXX.tgz openbsd
13
14 mkhybrid -b openbsd/floppy34.fs \
15   -c boot.catalog \
16   -l -J -L -r \
17   -o livecd-obsd.iso openbsd
```

Código Fonte 2.2 – Gerar um *live-CD* OpenBSD com a estrutura siteXX.tgz

### 2.3.3 Simple-CDD

O Simple-CDD é a ferramenta oficial de geração de imagens para o Debian, apresentado na Subseção 2.1.2. O Simple-CDD atua como encapsulador de ferramentas usadas para criar uma imagem personalizada de acordo com a versão informada do Debian. É possível informar uma lista de pacotes e a ferramenta irá se encarregar de garantir a instalação dos mesmo na imagem que será gerada. Ele tem suporte para múltiplos perfis e scripts de pós-instalação personalizados (DEBIAN.ORG, 2007).

O Simple-CDD, assim como os anteriores, também é um utilitário que consiste de um conjunto de outros programas. Com os comandos apresentados na listagem 2.3 é possível criar uma imagem de instalação baseada na versão *Jessie* do Debian com uma série de pacotes adicionais instalados.

```
1 #!/bin/bash
2
3 # Em um sistema Debian based
4
5 apt-get install simple-cdd
6
7 mkdir ~/my-simple-cdd
8 cd ~/my-simple-cdd
9
10
11 /usr/bin/build-simple-cdd \
12   --profiles-udeb-dist jessie \
13   --debian-mirror http://ftp.br.debian.org/debian --dist jessie
14
15 mkdir profiles
16
17 for p in lista-de-pacotes
18 do echo $p >> profiles/NAME.packages
19 done
20
```

```
21 /usr/bin/build-simple-cdd --profiles NAME
```

Código Fonte 2.3 – Gerar um *live-CD* Debian com a versão Jessie

## 2.4 Semelhanças

As três ferramentas apresentadas na Seção 2.3, possuem uma característica em comum. São ferramentas que aglomeram diversas outras para tornar possível a geração da imagem de instalação do *live-CD*.

Nesta Seção, serão apresentadas cada uma destas ferramentas e detalhado como as mesmas interagem para, ao final, resultar numa imagem de instalação.

### 2.4.1 syslinux

O Syslinux é uma coleção de gerenciadores de inicialização capazes de inicializar a partir de unidades de disco, CDs e através da rede. Alguns dos sistemas de arquivos suportados são FAT, ext2, ext3, ext4 e Btrfs de dispositivo único não compactados ([SYSLINUX.ORG](http://SYSLINUX.ORG), 2006).

### 2.4.2 isolinux

O Isolinux é um gerenciador de inicialização para Linux que opera com CD-ROMs ISO 9660, que rege o padrão de sistemas de arquivos para mídias de disco ótico ([ISO](http://ISO), 1988). Isso evita a necessidade de criar uma imagem de disco de emulação com espaço limitado para emulação de disquete ou problemas de compatibilidade emulação de disco rígido ([SYSLINUX.ORG](http://SYSLINUX.ORG), 2017).

Ou seja, o isolinux será responsável por complementar a funcionalidade do syslinux, permitindo que seja criada uma imagem iso híbrida que pode ser inicializada tanto de dispositivos óticos (CDs, DVDs, BlueRays) quanto mídias removíveis (*pen drive*, *microSD*, *SDCard*, etc.).

### 2.4.3 chroot

Um chroot é uma operação que altera o diretório raiz aparente do processo em execução atual e seus filhos. Um programa que é executado em um ambiente modificado não pode acessar arquivos e comandos fora dessa árvore de diretórios ambientais ([ARCHLINUX.ORG](http://ARCHLINUX.ORG), 1992).

Sua utilização se deve, pois caso um sistema seja tornado não inicializável, um chroot pode ser usado para voltar ao ambiente danificado após o carregamento de um sistema de arquivos raiz alternativo. No caso da geração do *live-CD*, ele é utilizado para

a partir do sistema de arquivos raiz do *live-CD* ser possível carregar o sistema de arquivos do SO que será instalado.

#### 2.4.4 mksquashfs

O mksquashfs é uma ferramenta responsável por criar um squashfs, que é um sistema de arquivos altamente compactado somente leitura para Linux. Ele usa um sistema de compactação que atua em arquivos, *inodes* e diretórios. Os *inodes* no sistema são muito pequenos e todos os blocos são compactados para minimizar a sobrecarga de dados. Tamanhos de bloco maiores que 4K são suportados até um máximo de 64K ([PAGES, 1996](#)).

Ela é a responsável por comprimir todos os arquivos que serão instalados no equipamento para que a imagem de instalação final possa ter um tamanho reduzido e não refletir o mesmo espaço ocupado no sistema de arquivos descompactado.

#### 2.4.5 xorriso

O xorriso copia objetos de arquivo de sistemas de arquivos compatíveis com POSIX em sistemas de arquivos ISO 9660 e permite a manipulação de tais sistemas de arquivos. Ele pode carregar as informações de gerenciamento de imagens ISO existentes e grava os resultados da sessão em mídia ótica ou em objetos do sistema de arquivos ([GNU.ORG, 2008](#)).

Esta ferramenta é a responsável por organizar toda a estrutura que que foi montada até agora em um arquivo iso que atenda às normal definidas na ISO 9660.



## 3 Metodologia

A fim de responder às questões de pesquisa levantadas, foram estabelecidas duas técnicas de investigação científica a fim de reunir estudos relevantes sobre as questões formuladas.

Por haver um extenso conjunto de material bibliográfico a respeito de alguns assuntos relevantes às QP's, como por exemplo as ferramentas utilizadas no processo, as normas da Organização Internacional para Padronização (ISO<sup>12</sup>) e Pedidos de Comentários (RFC<sup>13</sup>). Este referencial teórico pôde ser encontrado facilmente em bases de dados científicas, que são basicamente um aglomerado de centenas de revistas científicas em um único site, evitando assim o desperdício de tempo por parte do pesquisador.

Embora essa primeira parte da pesquisa tenha sido de fácil acesso, devido a grande disponibilidade acima citada, a segunda parte da pesquisa não retornou nenhum resultado válido nas bases utilizadas. Esta segunda parte, que trata de **Qual a melhor abordagem para a geração de um *live-CD* de instalação de Sistema Operacional Linux?**, praticamente não possui correspondências nas bases de dados. Isso se deve ao fato de que cada um dos distribuidores de mídias de instalação Linux prefere uma abordagem própria muito restrita ao seu contexto de trabalho.

### 3.1 Revisão Sistemática de literatura

A técnica utilizada para pesquisa da primeira parte foi a realização de uma revisão sistemática de literatura (RSL). Uma técnica de pesquisa que utiliza métodos sistemáticos para coletar dados, avaliar e sintetizar estudos de pesquisa. Nas revisões sistemáticas de literatura são formuladas questões de pesquisa de âmbito amplo ou restrito, que identificam e sintetizam estudos que se relacionam diretamente com o tema de revisão sistemática.

Embora a técnica de Revisão Sistemática tenha sido utilizada no presente trabalho, o objetivo do mesmo não é o de ser uma RSL, já que segundo (KITCHENHAM et al., 2009) seria necessário apresentarmos todos os refinamentos da string de busca utilizada na base de pesquisa. A listagem do artigos encontrados, os que foram excluídos e os que foram incluídos. O objetivo do uso da técnica foi somente o de ajudar a levantar referencial para o conhecimento do tema, que no início do trabalho era escasso.

Assim, a estratégia de pesquisa consistiu de uma busca automatizada na base de

---

<sup>12</sup> <<https://www.iso.org/about-us.html>>

<sup>13</sup> <<https://www.ietf.org/standards/rfcs/>>

pesquisa Scopus<sup>14</sup>. Foi definida a base de pesquisa Scopus, pois ela indexa bases importantes na área de Ciência da Computação, como IEEE, ACM, Elsevier. Além de ser a maior base de dados de citações e resumo de literatura revisada por pares: revistas científicas, livros e conferências<sup>15</sup>.

Com base nas buscas, os resultados encontrados foram submetidos aos critérios de inclusão e exclusão, que estão detalhados abaixo, e, então, um grupo inicial de artigos foi definido para a leitura. Visando atingir um número maior de artigos relevantes, também foi aplicada a técnica de *snowballing* para trás, visto que sugere-se que além da revisão sistemática, outras técnicas de pesquisa também são necessárias para atingir a melhor população de artigos possível (KITCHENHAM et al., 2009).

Assim, foi feita a análise das referências do grupo inicial de artigos a fim de encontrar novos artigos que respondessem às questões de pesquisa. Porém, o *snowballing* para trás foi aplicado apenas em um nível.

Por meio das especificações acima descritas os seguintes passos foram executados na realização da pesquisa:

1. Busca automatizada nas bases utilizando strings de busca. Obs: nos casos em que artigos repetidos foram encontrados entre as bases, apenas a primeira ocorrência foi considerada, sendo as demais descartadas;
2. Análise dos títulos do artigo e do *abstract*, a fim de verificar se há relação do mesmo com as QP's.
3. Análise integral dos artigos remanescentes, sendo descartados os artigos incompatíveis com as QP's;
4. *Snowballing* de um nível para trás dos artigos inclusos;
5. Aplicar passos 2 e 3 aos artigos encontrados por meio do passo 4.

## 3.2 Pesquisa Bibliográfica

A segunda técnica utilizada foi a pesquisa bibliográfica. Segundo (GIL, 2010) A pesquisa bibliográfica é desenvolvida com base em material já elaborado, constituído principalmente de livros e artigos científicos a fim de proporcionar maior familiaridade com o problema.

---

<sup>14</sup> <[www.scopus.com/home.uri](http://www.scopus.com/home.uri)>

<sup>15</sup> <<https://www.elsevier.com/americalatina/pt-br/scopus>>

Como o número de artigos científicos relacionados à segunda QP é escasso a pesquisa bibliográfica será baseada em manuais<sup>16</sup>, wikis<sup>17,18</sup>, as listas de email<sup>19</sup>, e o próprio código fonte<sup>20,21</sup>. A partir destas fontes podemos aumentar a familiaridade com o problema e chegar a uma solução que atenda satisfatoriamente o mesmo.

### 3.3 Escolha de Abordagem

A partir da consulta à literatura foi possível constatar que existe material suficiente para se gerar um *live-CD* de instalação de um Sistema Operacional Linux. Portanto, existem ferramentas que auxiliam no processo de geração de imagens de instalação. E estas ferramentas são na verdade facilitadores que agrupam um conjunto de outras ferramentas, já apresentadas neste trabalho, e que com as particularidades de organização apresentadas de cada uma das distribuições é possível gerar imagens personalizadas de instalação Linux daquela distribuição específica.

Embora haja um pouco de subjetivismo na pergunta Qual a melhor abordagem para a geração de um *live-CD* de instalação de Sistema Operacional Linux?, visto que o conceito de melhor é variável, para este trabalho será definido que a melhor abordagem diz respeito à facilidade de uso e a portabilidade da solução.

Nas três ferramentas, apresentadas na Seção 2.3, existem particularidades muito intrínsecas à distribuição que a ferramenta pertence. Desta forma, nosso quesito de portabilidade da solução não é atendido por nenhuma das três ferramentas.

Com relação ao quesito facilidade de uso, a ferramenta que mais preza por esta filosofia foi o Simple-CDD. Embora o mesmo seja altamente extensível e permita alterações variadas nas configurações do processo de instalação, estas configurações se restringem à distribuição Debian, caindo novamente no problema de portabilidade da solução.

Para alcançar o objetivo estabelecido neste trabalho, definiu-se a utilização das ferramentas apresentadas na Seção 2.4, que compõem as ferramentas utilizadas pelas distribuições, e a partir delas criar uma ferramenta própria. Além disso, estabeleceu-se o desenvolvimento de uma aplicação em Java, responsável por realizar o processo de instalação, baseada na filosofia do Simple-CDD de ser amigável ao usuário.

---

<sup>16</sup> <<https://www.debian.org/releases/stable/i386>>

<sup>17</sup> <<https://wiki.archlinux.org/index.php/Archiso>>

<sup>18</sup> <<https://wiki.debian.org/DebianInstaller>>

<sup>19</sup> <<https://lists.debian.org/debian-boot/>>

<sup>20</sup> <<https://git.archlinux.org/archiso.git>>

<sup>21</sup> <<https://salsa.debian.org/installer-team>>

## 4 Resultados

Neste Capítulo, serão apresentados os resultados obtidos durante a fase de desenvolvimento do trabalho. Conforme definido na Subseção 3.3 o desenvolvimento se dividiu em duas etapas. A primeira, consistiu em utilizar-se das ferramentas apresentadas na Seção 2.4 para criar uma ferramenta responsável por gerar *live-CDs* de forma rápida e automatizada. A segunda etapa foi referente ao desenvolvimento de uma aplicação Java para realizar o processo de instalação automatizada.

### 4.1 Ferramenta GeralSO

Por questões de escopo e de tempo, será utilizado o Sistema Operacional Debian Jessie para a geração do *live-CD* e do Sistema Operacional que irá ser instalado. Entretanto, os comandos apresentados aqui podem ser utilizados, sem nenhuma perda de generalidade, em qualquer outro sistema *Unix-Like* que possua disponíveis os pacotes das ferramentas apresentadas na Seção 2.4.

O primeiro passo é gerar a estrutura necessária para que seja possível gerar os *live-CD*. Para isso, foi escrito o *script* 4.1 que instala todos os pacotes necessários e organiza a estrutura de arquivos necessárias para iniciar o *live-CD*.

Dentro do diretório SO é onde ficará armazenado os arquivos do Sistema Operacional. Como em sistema *Unix-Like* tudo são arquivos (TORVALDS, 2002) não há nenhum problema manter os arquivos do Sistem Operacional dentro de outro SO. E com esta abordagem é possível mantermos qualquer distribuição que obedeça a esta filosofia.

```

1 #!/bin/bash
2
3 apt install debootstrap \
4   syslinux \
5   syslinux-common \
6   isolinux \
7   squashfs-tools \
8   xorriso
9
10 mkdir -p livecd/{chroot,SO} && cd livecd
11
12 debootstrap --arch=amd64 \
13   --variant=minbase jessie \
14   chroot http://ftp.br.debian.org/debian
15
16 chroot chroot

```

```
17
18 mount none -t proc /proc
19 mount none -t sysfs /sys
20 mount none -t devpts /dev/pts
21
22 echo 'Live-CD' > /etc/hostname
23 echo '127.0.0.1 localhost' > /etc/hosts
24 echo '127.0.0.1 DebianLive' >> /etc/hosts
25 echo ':::1 localhost ip6-localhost ip6-loopback' >> /etc/hosts
26 echo 'ff02::1 ip6-allnodes' >> /etc/hosts
27 echo 'ff02::2 ip6-allrouters' >> /etc/hosts
28 echo 'export LANG="C"' >> /etc/bash.bashrc echo 'export LC_ALL="C"' >> /etc
    /bash.bashrc
29
30 apt install -y linux-image-3.16.0-4-amd64
31
32 apt install -y live-boot \
33     systemd-sysv lvm2 cryptsetup libcryptsetup-dev \
34     xserver-xorg-core xserver-xorg xinit xterm \
35     ssh ifupdown net-tools \
36     i3wm isc-dhcp-client feh parted \
37     xz-utils bash-completion iputils-ping kbd
38
39 umount /proc /sys /dev/pts
40
41 exit
42
43 mkdir -p image/{live,isolinux}
44 cp chroot/boot/vmlinuz* image/live/vmlinuz
45 cp chroot/boot/initrd* image/live/initrd
46 cp chroot/boot/memtest86+.bin image/live/memtest86+.bin
47 cp chroot/boot/memtest86+_multiboot.bin image/live/memtest86+_multiboot.bin
48 cp /usr/lib/ISOLINUX/isolinux.bin image/isolinux/
49 cp /usr/lib/syslinux/modules/bios/* image/isolinux/
50
51 cat << EOF > image/isolinux/isolinux.cfg
52 UI menu.c32
53 prompt 0 menu title Live-CD
54 timeout 30
55 label Debian Live menu label ^Debian Live menu default kernel /live/vmlinuz
    append initrd=/live/initrd boot=live
56 label hdt menu label ^Hardware Detection Tool (HDT) kernel hdt.c32 text
    help HDT displays low-level information about the systems hardware.
    endtext
57 label Memtest86+ menu label ^Memory Failure Detection (memtest86+)
58 /live/memtest86+.bin
59 label Memtest86+ (multiboot) menu label ^Memory Failure Detection (
```

```
memtest86+) /live/memtest86+_multiboot.bin
60 EOF
61
62 cat << EOF > SO/lista-de-exclusao.txt
63 echo imagem.tar.gz
64 imagem.tar.xz
65 media
66 dev
67 proc
68 tmp
69 sys
70 run
71 EOF
```

Código Fonte 4.1 – Script para organizar a estrutura

Uma vez que a estrutura está organizada e os arquivos necessários do Sistema Operacional já estejam devidamente salvos no diretório SO, é possível executar o *script 4.2*. Que resultará na criação do arquivo *live-cd.iso* contendo um arquivo *imagem.tar.xz* comprimido do Sistema Operacional.

```
1 #!/bin/bash
2
3 tar cvJf chroot/root/imagem.tar.xz . -X lista-de-exclusao.txt
4
5 mksquashfs chroot image/live/filesystem.squashfs -comp xz -e boot
6
7 xorriso -as mkisofs -r -J -joliet-long \
8 -l -cache-inodes -isohybrid-mbr /usr/lib/syslinux/isohdpx.bin \
9 -partition_offset 16 -A "Live-CD" \
10 -b isolinux/isolinux.bin \
11 -c isolinux/boot.cat \
12 -no-emul-boot -boot-load-size 4 -boot-info-table \
13 -o live-cd.iso image
```

Código Fonte 4.2 – Ferramenta automatizada para Gerar um *live-CD*

## 4.2 Instalador Java

Foi desenvolvido um instalador em Java, responsável por executar todo o processo de instalação do SO de forma semiautomizada. O processo de instalação não pôde ser totalmente automatizado, pois há a necessidade de que o nome do equipamento seja informado durante a instalação. Por conta disto o processo de instalação é considerado semiautomatizado.

A Figura 1 apresenta o diagrama de classes do Instalador.

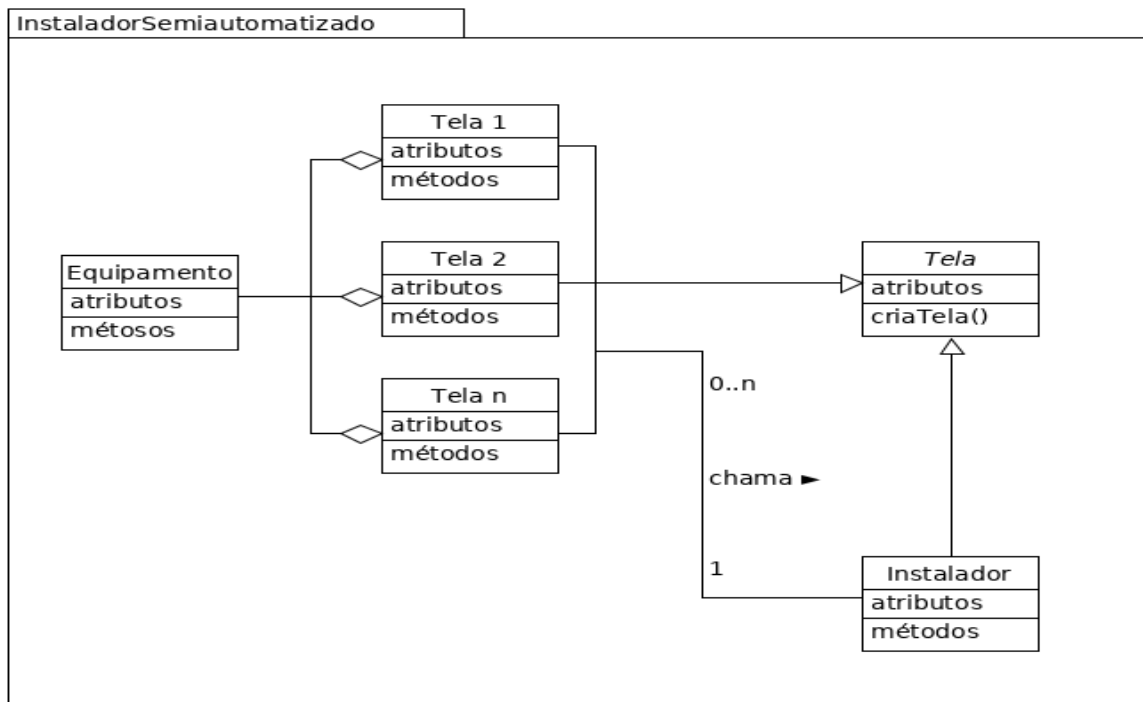


Figura 1 – Diagrama de Classes do Instalador Java.


### 4.3 Homologação do Instalador

Uma vez que o Instalador Java é responsável basicamente por chamadas de aplicativos do Sistema Operacional, os testes de validação do funcionamento das chamadas foram executados em laboratórios em equipamentos de teste. Visto que seria demandado um grande esforço para criação de testes unitários ou de mock para validar ocorrências no SO, como por exemplo testar se um comando responsável por zerar o disco rígido do equipamento, no início da instalação, ocorreu corretamente.

## 5 Conclusões

### 5.1 Instância de Solução

Durante a execução do instalador, há duas telas principais. A Figura 2 apresenta a tela que solicita as informações necessárias para dar prosseguimento ao processo de instalação.



The screenshot shows a Java installer window with a dark blue background. At the top, there are two radio buttons: 'Produção' (selected) and 'Homologação'. Below them, the text reads 'Entre com o Nome do Equipamento ex.: (UF123456TFLXX)'. There are two input fields: 'Nome Lógico:' with the value 'TFL' and 'Modelo:' with the value '4000'. A 'Continuar' button is at the bottom.

Figura 2 – Tela de Informações do Instalador Java.

Em seguida, ao clicar no botão de Continuar, todos os processos desde a limpeza do disco rígido apagando todos os dados do mesmo até a cópia dos arquivos do novo SO são feitos de forma automatizada. Todas as informações de progresso são exibidas na tela apresentada na Figura 3.



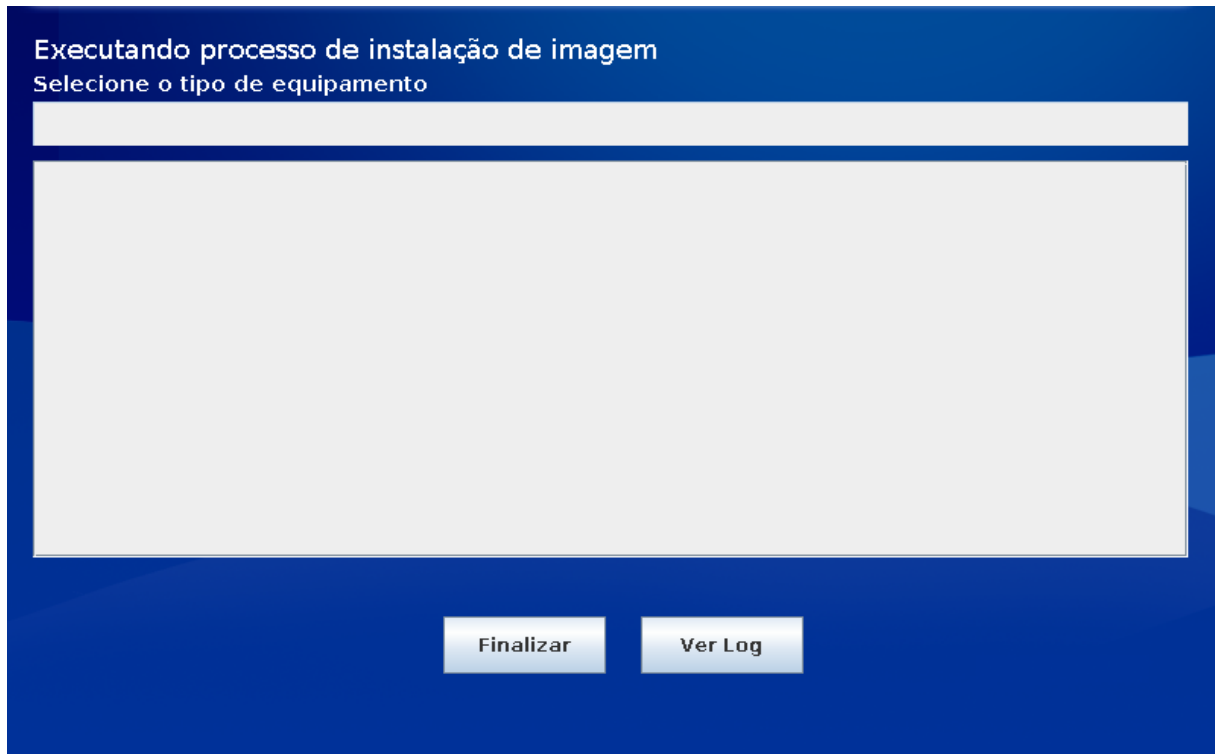


Figura 3 – Tela de Progresso do Instalador Java.

## 5.2 Possíveis aplicações

A solução apresentada neste trabalho pode ser utilizada por qualquer organização ou usuário que tenha como necessidade automatizar o processo de instalação de Sistemas Operacionais *Unix Like* em equipamentos que possuem características e configurações muito semelhantes.

## 5.3 Trabalhos futuros

Por questões de escopo e tempo só foi possível criar uma mídia de instalação utilizando o Sistema Operacional Debian. Como trabalhos futuros, é possível expandir os resultados para englobar também os demais Sistemas Operacionais, *OpenBSD* e *ArchLinux*, apresentado no decorrer do presente trabalho.

Além de adicionar estudos sobre o tema de variabilidade em linha de produto de software, a fim de que seja possível estabelecer modelos que prevêm as características e detalhes dos equipamentos onde os *live-CDs* serão instalados e conseqüentemente gerar mídias de instalação menores e mais rápidas de serem instaladas.

## Referências

- ARCHLINUX.ORG. *chroot*. 1992. Disponível em: <<https://wiki.archlinux.org/index.php/Chroot>>. Acesso em: 19/10/2018. Citado na página 22.
- ARCHLINUX.ORG. *Archiso*. 2001. Disponível em: <<https://wiki.archlinux.org/index.php/archiso>>. Acesso em: 19/09/2018. Citado na página 20.
- ARCHLINUX.ORG. *A simple, lightweight distribution*. 2001. Disponível em: <<https://www.archlinux.org/about>>. Acesso em: 19/10/2018. Citado na página 17.
- AUTHORITY, S. Q. *Installation Methods*. 2006. Disponível em: <[https://www.sqa.org.uk/e-learning/COS103CD/page\\_04.htm](https://www.sqa.org.uk/e-learning/COS103CD/page_04.htm)>. Acesso em: 09/12/2018. Citado na página 12.
- CLONEZILLA. *Clonezilla Live - A small bootable GNU/Linux distribution for imaging and cloning*. 2004. Disponível em: <<https://clonezilla.org/clonezilla-live.php>>. Acesso em: 01/10/2018. Citado na página 16.
- CROWLEY, E. Running linux in a windows computer lab. 2009. Disponível em: <[www.scopus.com](http://www.scopus.com)>. Citado na página 15.
- DEBIAN.ORG. *Debian, The Universal Operating System*. 1993. Disponível em: <<https://www.debian.org/>>. Acesso em: 19/10/2018. Citado 2 vezes nas páginas 17 e 18.
- DEBIAN.ORG. *Simple-CDD*. 2007. Disponível em: <<https://wiki.debian.org/Simple-CDD>>. Acesso em: 19/09/2018. Citado na página 21.
- FOUNDATION, I. F. S. *What is Free Software*. 2018. Disponível em: <<https://www.fsf.org/about/what-is-free-software>>. Acesso em: 17/11/2018. Citado 2 vezes nas páginas 15 e 16.
- GIL, A. C. *Como elaborar projetos de pesquisa*. [S.l.]: Atlas, 2010. Citado na página 25.
- GNU.ORG. *GNU xorriso*. 2008. Disponível em: <<https://www.gnu.org/software/xorriso/>>. Acesso em: 19/10/2018. Citado na página 23.
- GPARTED. *GNOME Partition Editor*. 2004. Disponível em: <<https://gparted.org/livecd.php>>. Acesso em: 01/10/2018. Citado na página 16.
- ISO. *Volume and file structure of CD-ROM for information interchange*. 1988. Disponível em: <<https://www.iso.org/obp/ui/#iso:std:iso:9660:ed-1:v1:en>>. Acesso em: 19/10/2018. Citado na página 22.
- KITCHENHAM, B. et al. Systematic literature reviews in software engineering - a systematic literature review. *Inf. Softw. Technol.*, Butterworth-Heinemann, Newton, MA, USA, v. 51, n. 1, p. 7–15, jan. 2009. ISSN 0950-5849. Disponível em: <<http://dx.doi.org/10.1016/j.infsof.2008.09.009>>. Citado 2 vezes nas páginas 24 e 25.

- OPENBSD.ORG. *Customizing the Install Process*. 1997. Disponível em: <<https://www.openbsd.org/faq/faq4.html#site>>. Acesso em: 19/10/2018. Citado na página 20.
- OPENBSD.ORG. *OpenBSD*. 1997. Disponível em: <<https://www.openbsd.org/>>. Acesso em: 19/10/2018. Citado 2 vezes nas páginas 17 e 18.
- OUTIN, N. *live-CD List*. 2017. Disponível em: <<http://livecdlist.com>>. Acesso em: 17/11/2018. Citado na página 15.
- PAGES, L. M. *mksquashfs (1) - Linux Man Pages*. 1996. Disponível em: <<https://www.systutorials.com/docs/linux/man/1-mksquashfs/>>. Acesso em: 19/10/2018. Citado na página 23.
- SILBERSCHATZ, A.; GALVIN, P. B. *Operating system concepts*. 9 edition. ed. [S.l.]: Wiley, 2014. Citado na página 12.
- SYSLINUX.ORG. *The Syslinux Project*. 2006. Disponível em: <[https://www.syslinux.org/wiki/index.php?title=The\\_Syslinux\\_Project](https://www.syslinux.org/wiki/index.php?title=The_Syslinux_Project)>. Acesso em: 19/10/2018. Citado na página 22.
- SYSLINUX.ORG. *What is ISOLINUX*. 2017. Disponível em: <<https://www.syslinux.org/wiki/index.php?title=ISOLINUX>>. Acesso em: 19/10/2018. Citado na página 22.
- SYSTEMRESCUE. *About SystemRescueCd*. 2006. Disponível em: <<http://www.system-rescue-cd.org/>>. Acesso em: 01/10/2018. Citado na página 16.
- TORVALDS, L. *Linux Mail List*. 2002. Disponível em: <[https://yarchive.net/comp/linux/everything\\_is\\_file.html](https://yarchive.net/comp/linux/everything_is_file.html)>. Acesso em: 19/10/2018. Citado na página 27.
- UTUNI, V. H. S. et al. Kcrystal: Linux 'live-cd' for powder crystallography. *Journal of Applied Crystallography*, v. 38, n. 4, p. 706–707, 2005. Disponível em: <[www.scopus.com](http://www.scopus.com)>. Citado na página 15.