



UNIVERSIDADE EDUARDO MONDLANE
FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA
CURSO DE ENGENHARIA ELECTRÔNICA

Relatório do Estágio Profissional

**IMPLEMENTAÇÃO DO SISTEMA DE EMPACOTAMENTO DE
COMPRIMIDOS *VICPACK* NO HOSPITAL CENTRAL DE
MAPUTO**

Autor: Edílio António Langa

Supervisor da Faculdade: Eng.º Omar Anlaue

Supervisor da Instituição (HCM): Eng.º Lisboa Chimuca

MAPUTO, Dezembro de 2019



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELECTROTÉNICA

CURSO DE ENGENHARIA ELECTRÔNICA

Relatório do Estágio Profissional

**IMPLEMENTAÇÃO DO SISTEMA DE EMPACOTAMENTO DE
COMPRIMIDOS *VICPACK* NO HOSPITAL CENTRAL DE
MAPUTO**

Autor: Edílio António Langa

Supervisor da Faculdade: Eng.º Omar Anlaue

Supervisor da Instituição (HCM): Eng.º Lisboa Chimuca

MAPUTO, Dezembro de 2019



UNIVERSIDADE EDUARDO MONDLANE
FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELECTROTÉNICA

AVALIAÇÃO DOS SUPERVISORES

Autor: Edílio António Langa

**IMPLEMENTAÇÃO DO SISTEMA DE EMPACOTAMENTO DE
COMPRIMIDOS VICPACK NO HOSPITAL CENTRAL DE
MAPUTO**

Supervisor da Faculdade

Nota

(Eng.º Omar Anlaue)

Supervisor da Instituição (HCM)

Nota

(Eng.º Lisboa Chimuca)



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELECTROTÉNICA

TERMO DE ENTREGA DE RELATÓRIO DO ESTÁGIO PROFISSIONAL

Declaro que o estudante: _____

Entregaram no dia ___/___/20__ as ___ cópias do relatório do seu Projecto do Curso
com a referência: _____

intitulado:

Maputo, ___ de _____ de 20__

O Chefe de Secretaria

DEDICATÓRIA

Este trabalho é dedicado ao Hospital Central de Maputo, pelo facto de desempenhar um papel fundamental no cuidado de vidas, sendo este o seu maior valor. É dedicado ao sector da manutenção hospitalar, sector este que tem empreendido esforços na reparação dos equipamentos necessários para o bom funcionamento do Hospital.

À Faculdade de Engenharia da Universidade Eduardo Mondlane, por ter sido muito útil no processo da minha formação intelectual.

Dedico por último, mas não menos importante, à cada Docente que contribuiu de forma activa, partilhando o seu conhecimento teórico e prático com objectivo de nos tornar Engenheiros com a capacidade de aproveitar os recursos existentes para responder as necessidades da humanidade.

AGRADECIMENTOS

Agradeço antes de tudo a Deus pelo dom da vida e por acompanhar todo meu percurso estudantil até a realização deste trabalho. Aos meus pais António Langa e Elisa Savela por terem investido humildemente para a minha formação.

Agradeço imenso a minha amiga de coração e companheira das batalhas do reino, Daniel Marta Joane que no dia 22 de maio de 2017 me incentivou a não desistir da minha formação académica.

Agradeço aos Docentes da Faculdade de Engenharia pelos desafios colocados ao longo da formação, criando assim nos estudantes um espírito de auto-superação e a capacidade de trabalhar sobre pressão.

Agradeço ao Senhor Basílio Pinoca, técnico de manutenção hospitalar, por ter me apresentado o desafio de tentar solucionar o problema de empacotamento que vigora na instituição.

Por fim agradeço aos meus supervisores, Engenheiro Omar Anlaue e Engenheiro Lisboa Chimuca, que acompanharam o desenvolvimento deste projeto, apresentado sugestões que foram muito uteis.

“Filho, há mais uma coisa que eu quero dizer: os livros sempre continuarão a ser escritos; estudar de mais cansa a mente”

(Rei Salomão)

RESUMO

O presente relatório visa descrever as actividades realizadas durante o estágio profissional no Hospital Central de Maputo, tendo como o principal objectivo o estudo da possibilidade da implementação de um sistema automático de empacotamento de comprimidos no mesmo.

O relatório está estruturado em cinco capítulos, sendo o primeiro capítulo a parte introdutória, contendo informações relativas ao relatório, a justificativa, os objectivos e a metodologia usada. O segundo capítulo visa fazer a apresentação da instituição e descrever as actividades realizadas pelo autor, o terceiro capítulo apresenta a formulação do problema, soluções propostas, resumo teórico, culminando com a descrição da solução escolhida.

No quarto capítulo trata-se da implementação do projecto, onde são apresentadas as simulações e os testes realizados e no último capítulo são apresentadas as considerações finais do trabalho, as conclusões e recomendações. Também foram incluídos anexos que contêm especificações técnicas e informações adicionais.

O presente relatório justifica-se pelas várias tentativas enfrentadas pelo hospital de implantar o sistema de empacotamento *VICPACK* na sua farmácia, contudo sem sucesso. A implementação desse sistema é de extrema importância e urgência, pelo facto do nível da demanda de comprimidos aumentar a cada ano e o sistema de empacotamento manual se tornar cada vez mais trabalhoso e desgastante.

Para o alcance dos objectivos pré-estabelecidos, foi usada a metodologia mista, que envolve o método qualitativo e quantitativo e por meio das simulações e testes realizados concluiu-se que o sistema pode ser implementado.

ESTRUTURA DO RELATÓRIO

O relatório está estruturado em cinco capítulos, nomeadamente:

- **Capítulo 1:** este capítulo é a parte introdutória, contém informações relativas ao relatório, a justificativa, os objectivos e a metodologia usada;
- **Capítulo 2:** o segundo capítulo visa fazer a apresentação da instituição e descrever as actividades realizadas pelo o autor durante o estágio profissional;
- **Capítulo 3:** o terceiro capítulo apresenta a formulação do problema, soluções propostas, resumo teórico, culminando com a descrição da solução escolhida;
- **Capítulo 4:** neste capítulo se trata da implementação do projecto. Aqui, mostram-se as simulações e os testes realizados;
- **Capítulo 5:** sendo este o ultimo capítulo são apresentadas as considerações finais do trabalho, conclusões, recomendações e as referências bibliográficas;

Também foram incluídos anexos que contêm especificações técnicas e informações adicionais.

ABREVIATURAS E SIGLAS

HCM- Hospital Central de Maputo

SMH- Sector de Manutenção Hospitalar

USB- *Universal serial Base*

IEEE- Instituto de Engenheiros Eléctricos e Electrónicos.

TOF- Topo do formulário

ÍNDICE

INTRODUÇÃO	1
1.1. NOTA INTRODUTÓRIA	2
1.2. JUSTIFICATIVA	2
1.3. OBJECTIVOS.....	3
1.3.1. Objectivo Geral	3
1.3.2. Objectivos específicos	3
1.4. METODOLOGIA.....	4
ACTIVIDADES REALIZADAS	5
2.1. NOTA INTRODUTÓRIA	6
2.2. DESCRIÇÃO DA INSTITUIÇÃO	6
2.2.1. BREVE HISTORIAL DO HOSPITAL.....	7
2.2.2. ANTIGOS DIRECTORES DO HOSPITAL CENTRAL DE MAPUTO	8
2.2.3. DIRECÇÃO ADMNISTRATIVA.....	8
2.2.4. SECTORES.....	9
2.3. ACTIVIDADES REALIZADAS	11
ESTUDO DO SISTEMA	13
3.1. FORMULAÇÃO DO PROBLEMA.....	14
SOLUÇÕES.....	14
3.2. ANÁLISE DAS SOLUÇÕES.....	14
3.3. DESENVOLVIMENTO DA SOLUÇÃO PROPOSTA	15
3.3.1. MÁQUINA EMBALADORA HF200	16
3.3.1.1. Função do equipamento.....	16
3.3.1.2. Descrição das características gerais do equipamento	17
3.3.1.3. Funcionamento	22
3.3.1.4. Preparação para operar com o equipamento.....	23

3.3.1.5. Operação	23
3.3.1.6. Problemas enfrentados na tentativa do empacotamento	24
3.3.2. IMPRESSORA DE ETIQUETAS.....	27
3.3.3. PROGRAMA GERADOR DE ETIQUETAS.....	32
3.3.3.1. DESENVOLVIMENTO DO PROGRAMA GERADOR DE ETIQUETAS	34
SIMULAÇÕES E TESTES	39
4.1. PROGRAMA.....	40
4.2. IMPRESSORA.....	40
4.3. EMPACOTADORA	42
CONSIDERAÇÕES FINAIS	44
5.1. CONCLUSÃO.....	45
5.2. RECOMENDAÇÕES.....	45
5.3. BIBLIOGRAFIA	46
ANEXOS	47

ÍNDICE DE FIGURA

Figura 1: Diagrama de Blocos do Sistema	15
Figura 2: Máquina embaladora Viclina HF200A [5]	16
Figura 3: Vista lateral direita da Máquina embaladora Viclina HF200 [5]	18
Figura 4 : Vista lateral esquerda da Máquina embaladora Viclina HF200 [5]	19
Figura 5: Vista frontal da Máquina embaladora Vicilina HF 200 [8]	19
Figura 6: Esquema eléctrico Máquina embaladora Viclina HF200 [5]	21
Figura 7: papel com revestimento termo-soldável para a embalagem	26
Figura 9: Impressora Datamax M-Class Mark II	27
Figura 10: Painel de controle da impressora Datamax M-Class Mark II	27
Figura 11: Funções específicas do painel de controle da impressora Datamax M-Class Mark II	28
Figura 12: Conexão Paralela	30
Figura 13: Conexão USB	30
Figura 14: Conexão serial	30
Figura 15: Encaminhamento de fita com o lado do revestimento para dentro	31
Figura 16: Fluxograma do Processo	32
Figura 17: Fluxograma do programa	34
Figura 18: Tela do Login	35
Figura 19: Mensagem de erro de senha	35
Figura 20: Tela de boas vindas	36
Figura 21: Tela Principal	36
Figura 22: Tela de criação do modelo	37
Figura 23: Teste do programa gerador de etiquetas	40
Figura 24: Teste do posicionamento dos rolos na Impressora	40
Figura 25: Teste de impressão sem visualizar os dados inseridos	41

Figura 26: Teste de Impressão com múltiplas visualizações	41
Figura 27: Empacotamento com erro de corte e selagem.....	42
Figura 28: Ajuste da temperatura das mandibulas	42
Figura 29: Ajuste da altura da tesoura	42
Figura 30: Processo de selagem.....	43
Figura 31: Teste final da etiqueta	43
Figura 32: Teste final do empacotamento de comprimido.....	43

ÍNDICE DE TABELA

Tabela 1: Breve historial do HCM	7
Tabela 2: Directores do HCM.....	8
Tabela 3: Efectivos do SMH por carreira.....	9
Tabela 4: Equipamentos concertados no SMH	11
Tabela 5: Comparação das soluções.....	15
Tabela 6: medidas da Máquina embaladora Viclina HF200A.....	17
Tabela 7: Alimentação da Máquina embaladora Viclina HF200A	17
Tabela 8: Dimensões das bobines do material de embalagem.....	17
Tabela 9: Dimensões máximas da área de embalamento	18
Tabela 10: Itens exibidos no menu do usuário.....	28

CAPÍTULO 1

INTRODUÇÃO

No primeiro capítulo são apresentadas questões introdutórias, as quais envolvem a justificativa, os objectivos e a metodologia usada.

1.1. NOTA INTRODUTÓRIA

O ciclo de vida do ser humano é baseado no nascimento, crescimento, reprodução e morte. Nem sempre este ciclo se cumpre de forma sequencial, muitas vezes a morte tem se antecipado, chegando a ocorrer no processo do nascimento, crescimento e até mesmo no processo da multiplicação. No intuito de evitar a morte surge a medicina.

A medicina é área da ciência que tem como o seu maior valor a vida, sendo que esta subdivide-se em medicina tradicional e medicina moderna.

O presente relatório, visa descrever as actividades realizadas durante o estágio profissional no Hospital Central de Maputo, sendo esta uma instituição de medicina moderna e tem como objectivo principal implementar o sistema de empacotamento *VICPACK* na farmácia do mesmo hospital acima citado.

Este relatório foi desenvolvido no âmbito da culminação do curso de Licenciatura em Engenharia Electrónica através da cadeira de estágio profissional.

1.2. JUSTIFICATIVA

O tema escolhido para o presente relatório justifica-se pelas várias tentativas enfrentadas pelo hospital de implementar o sistema de empacotamento *VICPACK* na sua farmácia, contudo sem sucesso.

Sendo, o HCM o principal hospital do país, a implementação desse sistema é de extrema importância e urgência, pelo facto do nível da demanda de comprimidos aumentar a cada ano e o sistema de empacotamento manual se tornar cada vez mais trabalhoso e desgastante.

1.3. OBJECTIVOS

1.3.1. Objectivo Geral

- Estudar a possibilidade da implementação de um sistema automático de empacotamento de comprimidos no Hospital Central de Maputo.

1.3.2. Objectivos específicos

- Estudar o modo de funcionamento da máquina empacotadora *VICPACK*;
- Estabelecer a comunicação entre a máquina empacotadora *VICPACK* e a impressora *datamax-o'neil*;
- Desenvolver um programa usando a linguagem de programação java para registar os dados a serem impressos na embalagem;
- Testar o funcionamento do sistema e corrigir possíveis erros.

1.4. METODOLOGIA

Pela natureza do projecto a metodologia conveniente para este trabalho é a metodologia mista, que envolve o método qualitativo e quantitativo.

- Para o estudo do modo de funcionamento da máquina empacotadora *VICPACK* foi feita uma pesquisa qualitativa que consistiu em uma revisão bibliográfica baseada na literatura disponível (livros, páginas da internet, folhas de dados e vídeos);
- Para se estabelecer a comunicação entre a máquina empacotadora *VICPACK* e a impressora *datamax-o'neil* foram feitas pesquisas bibliográficas e posteriormente teste a vazio (com ausência da maquina empacotadora);
- Para desenvolver o programa usando a linguagem de programação java para registar os dados a serem impressos na embalagem foi usado o *NetBeans IDE 8.2*;
- Para validar o sistema foi usado o método quantitativo, onde o sistema foi submetido a simulações e testes no SMH.

CAPÍTULO 2

ACTIVIDADES REALIZADAS

Este capítulo visa fazer a apresentação da instituição e descrever as actividades realizadas.

2.1. NOTA INTRODUTÓRIA

O presente relatório é fruto do estágio realizado no hospital Central de Maputo no período de Setembro a Dezembro do ano em curso, sendo esta uma das formas de conclusão do curso. O autor neste período foi envolvido em diversas actividades nos diversos sectores que o hospital contém.

2.2. DESCRIÇÃO DA INSTITUIÇÃO

O Hospital Central de Maputo, é uma unidade sanitária de referencia nacional, que desenvolve actividades assistenciais, de formação e investigação, presta assistência a doentes de todo o país em quase todas as áreas médicas e cirúrgicas. Está localizado na Avenida Agostinho Neto nº 167 Caixa Postal 1164, Maputo-Moçambique

O Hospital Central de Maputo tem vários trabalhadores distribuídos por diferentes grupos profissionais, nomeadamente médicos, administrativos, seguranças, pessoal de áreas de apoio como cozinha, lavandaria, esterilização, entre outras, tendo como o atual Director Geral o Dr. Mouzinho Saíde.

Pelo seu tempo de existência o HCM necessita permanentemente de obras de manutenção e reabilitação para manter o seu padrão de qualidade e dignificar os seus utentes e trabalhadores. [6]

São também feitas constantemente instalações de novos equipamentos para responder as necessidades do desenvolvimento da medicina neste seculo.

Os grandes desafios que o HCM tem neste momento e durante os próximos anos é de se adaptar as necessidades dos seus utentes, dotando-se de meios humanos e materiais suficientes e adequados para um desempenho que se pretende de excelência. [6]

2.2.1. BREVE HISTORIAL DO HOSPITAL

Abaixo é apresentada a tabela do breve historial do hospital.

Tabela 1: Breve historial do HCM

ANO	FEITOS
1876	Autorização Governamental para a construção do Hospital
1877	Iniciam as obras de construção do Hospital de Lourenço Marques
1897	É designado de Rainha D. Amelia
1921	O Hospital toma a designação de Hospital Miguel Bombarda (HMB);
1975	Pelo Decreto 08/75 de 18 de Janeiro o Governo de Moçambique faz a fusão do HCMB e o Hospital da Universidade de Lourenço Marques formando o Hospital Central de Lourenço Marques que actualmente é denominado de Hospital Central de Maputo.

De acordo com os dados apresentados pode-se verificar que o HCM é uma instituição com mais de 100 anos de existência, tendo sofrido mudanças não simplesmente no nome mas também nas tecnologias usadas, surgindo assim a necessidade de um sector de manutenção com capacidade de trabalhar com variedade de tecnologias.

2.2.2. ANTIGOS DIRECTORES DO HOSPITAL CENTRAL DE MAPUTO*Tabela 2: Directores do HCM*

ANO	DIRECTOR GERAL
2017 até a Atualidade	Dr. Mouzinho Saíde
Abril 2012 – 2017	Dr. João Manuel De Carvalho Fumane
2005 – Abril 2012	Dr. Francisco José De Almeida Cândido
Abril 2001-2005	Dr. António Bonse Bomba
Outubro 1998 – Abril 2001	Dr. Rogério Prista Cunha
Junho 1996 – Setembro 1998	Dr. Ildefonso R. Munatatha
Out. 1990 – Dezembro 1994	Dr. Aurélio Amândio Zilhão
Outubro 1985 – Dezembro 1990	Dr. Orlando F. Da Silva Vieira
Jan. 1975 – Outubro 1985	Dr. Fernando Everard Do Rosário Vaz

2.2.3. DIRECÇÃO ADMINISTRATIVA

A Direcção Administrativa tem o seguinte papel:

- Garantir a segurança, e manutenção das instalações e boa conservação dos demais bens patrimoniais do Estado existentes no HCM;
- Coordenar a elaboração da proposta de planos de actividades da sua área e orçamento e submete-los à apreciação e aprovação do Conselho da Direcção;
- Colaborar com outras áreas com vista a optimização de recursos disponíveis e garantir a melhoria da qualidade dos serviços. [8]

2.2.4. SECTORES

O hospital Central de Maputo é composto por vários sectores, sendo o sector da Manutenção Hospitalar(SMH) o foco deste trabalho.

O Serviço de Manutenção Hospitalar está subdividido por diferentes sectores de actividades a saber: Sector de Administração, de Transporte, Electricidade, Electromedicina, Micromecânica, Refrigeração, Pintura, Carpintaria, Serralharia, Canalização, Construção civil, Armazém, Lavandaria, Vapor e Costura.

EFFECTIVO DO SMH POR CARREIRA

Tabela 3: Efectivos do SMH por carreira

CARREIRA	QUANTIDADE
Técnicos Superiores	01
Técnicos Médios de Manutenção	11
Técnicos Básicos de Manutenção	06
Electricistas Básicos	04
Técnicos Médios Administrativos	04
Técnico Básico Administrativo	11
Pintura	02
Carpintaria	02
Serventes	35
Motoristas	39
Lavandaria	45
TOTAL	160

ATRIBUIÇÕES DO SERVIÇO DE MANUTENÇÃO HOSPITALAR

Este serviço está vocacionado as seguintes reparações:




- Sistema e circuitos eléctrico, de frio/refrigeração e vapor;
- Equipamentos electrónicos em especial da área médica e de meios auxiliares de diagnóstico ou laboratorial,
- Candeeiros cialíticos e sistemas de gases medicinais;
- Carpintaria, serralharia, canalização, pintura, construção civil;
- Fiscalização de pequenas obras, lavandaria;
- Gestão de meios circulantes (transporte de doentes e bens).

2.3. ACTIVIDADES REALIZADAS

O sector de manutenção trabalha de acordo com as requisições, desta forma abaixo é apresentado em forma de tabela os principais equipamentos que tem sofrido avarias.

Tabela 4: Equipamentos concertados no SMH

Ordem	Designação	Imagem
1	Monitor digital de Pressão sanguínea	
2	Bomba de Infusão	
3	Siringa de Infusão	
4	Economizador de álcool	

5	Televisor	
6	Aspirador de secreções	
7	Aparelho de esterilização	

CAPÍTULO 3

ESTUDO DO SISTEMA

Neste capítulo é apresentada a formulação do problema, as possíveis soluções, as comparações, culminando com a descrição da solução escolhida;

3.1. FORMULAÇÃO DO PROBLEMA

Moçambique tem verificado um crescimento populacional, um grupo considerável de pessoas tem abandonado a medicina tradicional, substituindo-a pela medicina moderna. O Hospital Central de Maputo é a principal unidade hospital do país, junto ao aumento populacional o nível da demanda de comprimidos tem aumentado a cada ano, facto que contribui para que o sistema de empacotamento manual se torne cada vez mais trabalhoso e desgastante, surgindo assim o problema da **baixa eficiência no processo de empacotamento manual de comprimidos em doses unitárias.**

SOLUÇÕES

São abaixo listadas as propostas de solução para o problema apresentado:

Solução 1: Contratação de pessoal responsável pelo empacotamento manual dos comprimidos.

Solução 2: Aquisição de um novo equipamento de empacotamento automático de comprimidos.

Solução 3: Implementação do sistema de empacotamento *Vicpack* já existente.

3.2. ANÁLISE DAS SOLUÇÕES

Para a prossecução das suas actividades o HCM dispõe de fundos maioritariamente alocados pelo orçamento Geral do Estado e em menor volume de fundos próprios gerados pelo “atendimento especial e personalizado”. De um modo geral, pode-se dizer que estes meios estão muito aquém das necessidades dum hospital com a dimensão do HCM. [6]

Sendo assim, os critérios tomados como base para a escolha da solução são os seguintes:

1º. alta eficiência no processo de empacotamento;

2º. baixo custo financeiro;

A tabela que se segue apresenta uma comparação das soluções acima mencionadas.

Tabela 5: Comparação das soluções

Solução	1º Critério	2º Critério
Solução 1	X	X
Solução 2	✓	X
Solução 3	✓	✓

Tendo em conta a comparação feita, a melhor solução é a Implementação do sistema de empacotamento Vicpack já existente.

3.3. DESENVOLVIMENTO DA SOLUÇÃO PROPOSTA

Tendo sido escolhida a Implementação do sistema de empacotamento Vicpack como a melhor solução, segue-se uma análise mais detalhada e sistema. Abaixo são apresentados os blocos que compõem o sistema:



Figura 1: Diagrama de Blocos do Sistema

De acordo com o diagrama de blocos acima ilustrado pode-se verificar que o sistema é composto por três blocos funcionais, nomeadamente o Computador que por meio de um programa gera a etiqueta a ser impressa, que é responsável pela impressão da etiqueta e a Empacotadora, sendo esta a principal unidade, responsável pelo empacotamento do comprimido.

Para o melhor entendimento do sistema a análise do sistema será feita iniciando do último bloco para o primeiro.

3.3.1. MÁQUINA EMBALADORA HF200

A máquina encontra-se construída para o embalamento de elementos sólidos, líquidos e pós em geral, independentemente da natureza do material que compõe estes produtos. Os produtos a embalar podem ser especialidades de ferramentaria, produtos de laboratórios farmacêuticos, perfumaria, produtos de alimentação, etc. [5]

Para este projecto a maquina será usada para o empacotamento de produtos de natureza sólida, farmacêuticos, especificamente comprimidos.

Abaixo é apresentada a figura da Máquina embaladora *Viclina* HF200A:



Figura 2: Máquina embaladora *Viclina* HF200A [5]

3.3.1.1. Função do equipamento

A máquina encontra-se construída para o embalamento de medicamentos (pastilhas, cápsulas, drageias, etc.).

Apesar das eventuais aplicações anteriormente referidas, a flexibilidade de aplicações que permite a sua configuração, possibilita que este equipamento seja adaptado a qualquer outra aplicação de embalamento. Para tal é necessário trocar o

grupo de cabeças, mandíbulas, arrastos, ficando preparada para uma nova aplicação de embalagem com outro tipo de artigos

3.3.1.2. Descrição das características gerais do equipamento

Medidas

Tabela 6: medidas da Máquina embaladora Viclina HF200A

Medidas	HF-200
Altura	550 mm
Largura (máximo)	1100 mm
Profundidade	600 mm
Peso (sem embalagem)	115 Kg

Alimentação

Tabela 7: Alimentação da Máquina embaladora Viclina HF200A

Alimentação	HF-200
Tipo	Monofásica
Tensão	220 V
Potencia	950 W

Dimensões das bobines do material de embalagem

Tabela 8: Dimensões das bobines do material de embalagem

HF-200	
Largura máxima	76 mm
Diâmetro exterior máximo	220 mm
Diâmetro mandril	76 mm

Dimensões máximas da área de embalagem

Tabela 9: Dimensões máximas da área de embalagem

HF-200	
HF-200	76 mm X 65 mm.

Elementos

A embaladora *Viclina* contém os seguintes elementos:

- Regulador de temperatura, com sonda para controle de termo-selagem
- Variador electrónico de velocidade de embalagem
- Fotocélula para centrar os textos

A máquina HF-200 é composta por uma estrutura, que inclui o equipamento eléctrico e de manobra para assegurar a segurança do equipamento.

Na parte externa frontal, apresenta-se a localização dos órgãos e mecanismos que efectuam o embalagem do produto, nomeadamente:

- Cabeças porta bobinas (material termo-soldável)
- Grupo soldador (mandíbulas, rolos)

O grupo soldador e o grupo de corte, encontram-se protegidos por tampas de metacrilato transparente com interruptores de superfície que suspendem o processo de trabalho normal se forem transferidas ou eliminadas.

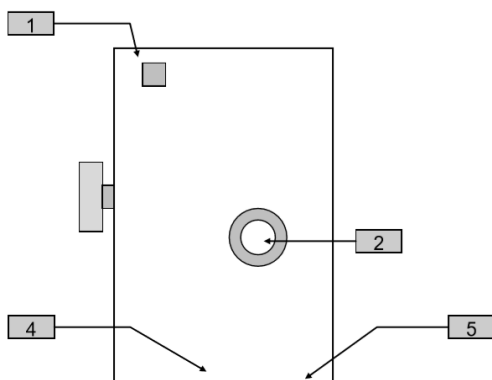


Figura 3: Vista lateral direita da Máquina embaladora *Viclina* HF200 [5]

- 1- Interruptor geral
- 2- Volante motor (para accionar à mão e confirmar o sentido de rotação, respeitando sempre a direcção indicada pela seta).
- 4- Conector para sinal de impressora.
- 5- Tomada 220V AC para a impressora (funciona ao conectar-se à máquina)

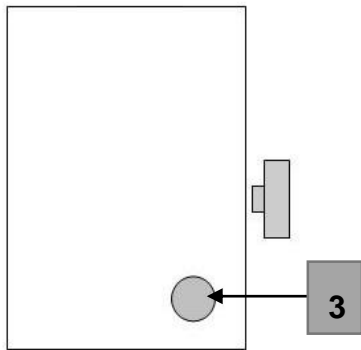


Figura 4 : Vista lateral esquerda da Máquina embaladora Viclina HF200 [5]

3- Regulador da altura das tesouras

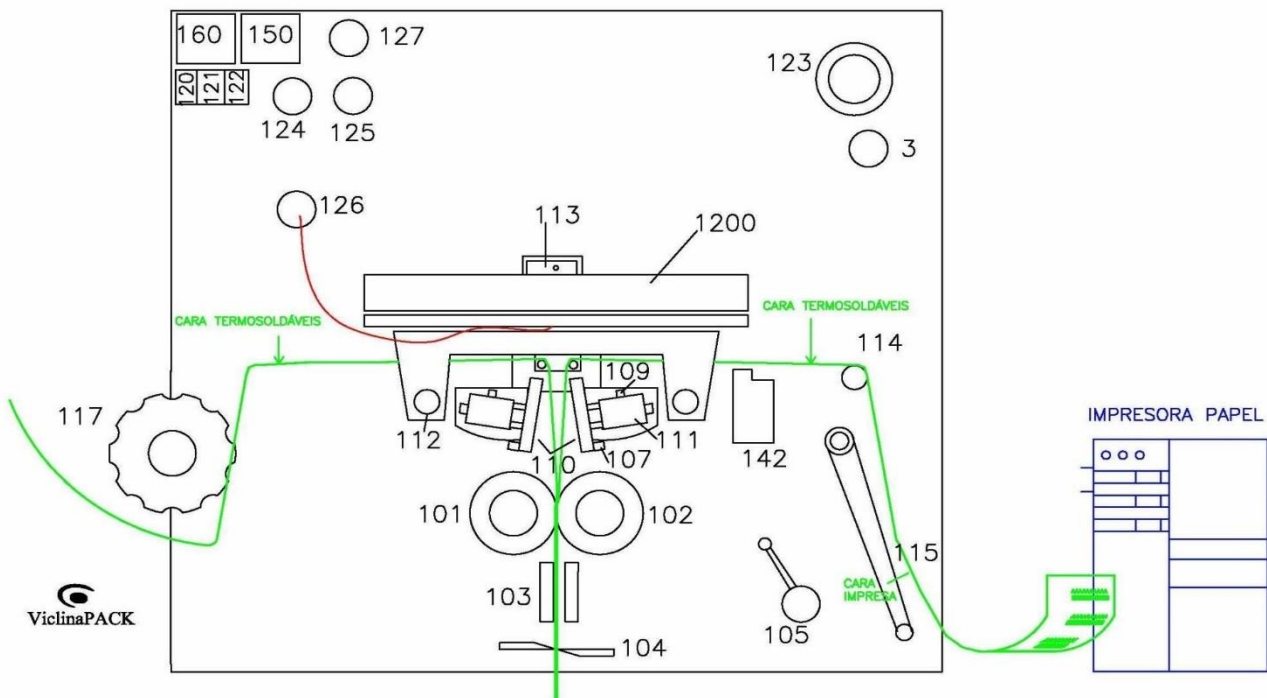


Figura 5: Vista frontal da Máquina embaladora Viclina HF 200 [8]

③ Botão corte tesoura

①①① Rolo principal de arrasto grafilado.

①①② Rolos de pressão de arrasto com aros de borracha.

①①③ Guia tesoura

①①④ Tesouras

①①⑤ Alavanca de manobra de rolos

①①⑦ Sonda reguladora de temperatura.

①①⑧ Parafuso para fixar a mandíbula.

①①⑨ Parafuso para o contacto eléctrico da resistência calefactora da mandíbula

①①⑩ Mandíbula de selagem.

①①① Braço porta mandíbulas.

①①② Guia para alimentador.

①①③ Caixa guia.

①①④ Alavanca guia.

①①⑤ Alavanca impressora.

①①⑦ Cubos porta bobinas de papel.

①②① Botão verde de marcha.

①②① Indicador de máquina a trabalhar.

①②② Botão vermelho de paragem.

①②③ Botão de paragem de emergência

①②④ Botão de inicio. Acende-se a luz azul quando tiver que ser pressionado.

①②⑤ Selector modo manual-automático

①②⑥ Conector cabo acessório para o modo Manual-Automático

①②⑦ Variador da velocidade do motor

①④② Focélula

①⑤① Regulador de temperatura da soldadura.

①⑥① Programador corte tesouras e contador de unidades embaladas

①②①① Bandeja blister

Abaixo é apresentado o esquema eléctrico da máquina:

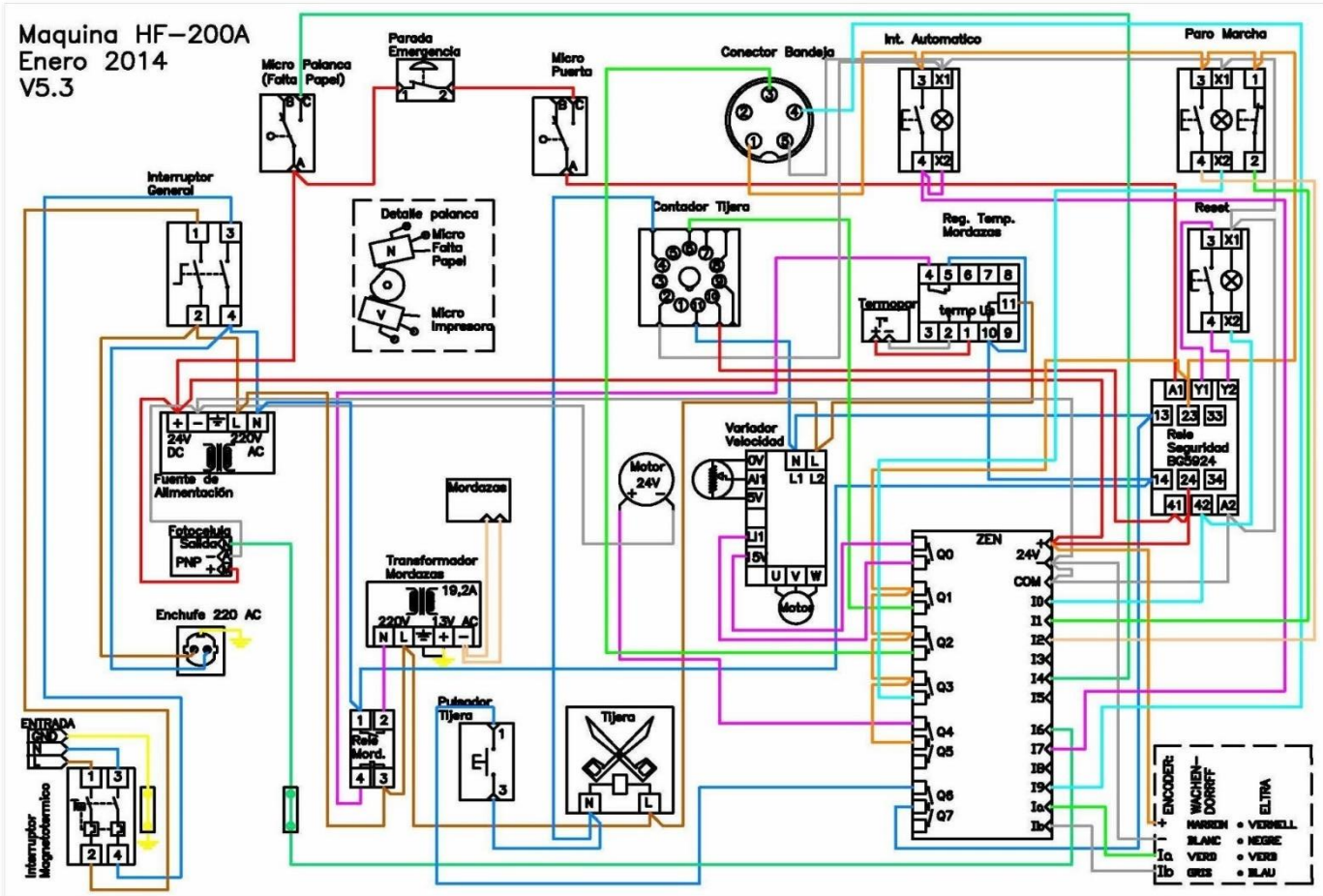


Figura 6: Esquema eléctrico Máquina embaladora Viclina HF200 [5]

3.3.1.3. Funcionamento

Ao se accionar o interruptor geral, o transformador e o circuito de controle e segurança são alimentados.

Deve-se confirmar que o botão de paragem de emergência (*stop*) não se encontra bloqueado.

Deve-se confirmar igualmente que as protecções exteriores estão correctamente colocadas. Se tudo estiver correcto, accionar-se-á o botão de inicio (luz azul) e a máquina passará a situação de aquecimento das mandíbulas, mas não mostrará nenhum tipo de movimento.

Uma vez aquecidas as mandíbulas estarão criadas as condições de trabalho e a maquina começará a funcionar, apertando o botão verde de marcha, iluminar-se-á o indicador de maquina em marcha.

Para aquecer as mandíbulas, selecciona-se a temperatura mediante os selectores correspondentes a cada dígito, acender-se-á uma luz vermelha no relógio indicando que as mandíbulas estão a aquecer. Só deve-se iniciar o empacotamento quando a temperatura for igual à programada. Sequencialmente e em intervalos irregulares esta luz acender-se-á e apagará de acordo com o esfriamento ou o aquecimento das mandíbulas.

POSIÇÃO DAS TESOURAS: ajusta-se a sua posição em altura mediante o botão de ajuste e comprova-se que a tesoura cortou na linha com pontos de separação do papel.

EMBALAMENTO: antes de carregar a máquina com a peças a embalar é necessário comprovar se a soldadura está correcta. Em cada bolsa deverá formar-se uma almofada de ar, cuja elasticidade nota-se ao tacto premindo-a com os dedos.

Carrega-se as embalagens e analisa-se se as primeiras unidades saem correctamente, ajustando se necessário a pressão dos rolos e posição das tesouras.

3.3.1.4. Preparação para operar com o equipamento

Para operar com a máquina deve-se seguir os seguintes passos:

1. Ligar a máquina à corrente.
2. Confirmar que o papel se encontra correctamente colocado na máquina.
3. Confirmar que as conexões entre a máquina e a impressora estão correctas.
4. Caso se utilize algum acessório confirmar que este também se encontra correctamente ligado.

3.3.1.5. Operação

1. Girar o interruptor geral situado no lado direito da máquina.
- ✓ Acender-se-á uma luz azul no botão de Início.
 2. Pressionar o botão Início.
- ✓ apagar-se-á a luz azul
 3. Esperar que a temperatura das mandíbulas esteja na temperatura adequada (aproximadamente 120°).
 4. Seleccionar o modo em que se vai utilizar a máquina: manual ou automático.
- ✓ Modo manual: a máquina funciona de maneira contínua sem parar e sem ter em conta se caiu ou não algum medicamento no recipiente;
- ✓ Modo automático: a máquina neste modo possui um sensor que detecta quando caiu algum medicamento e inicia posteriormente o ciclo para o seu embalamento. Se antes de terminar o ciclo existir algum medicamento bem posicionado, a máquina continua o embalamento sem parar. No entanto, caso não exista um medicamento bem posicionado, a máquina para o embalamento até que esta situação seja corrigida.
5. Pressionar o botão Marcha e a máquina procederá ao embalamento do produto.
6. Para parar a máquina é necessário pressionar o botão de paragem
7. Se durante o funcionamento da máquina ocorrer alguma anomalia ou incidente dever-se-á pressionar a paragem de emergência para parar a máquina imediatamente.

3.3.1.6. Problemas enfrentados na tentativa do empacotamento

Tendo em conta que o HCM teve várias tentativas de implementar o sistema, contudo sem sucesso, para melhor estudo do sistema foi feita uma análise em etapas, sendo a primeira etapa a análise da maquina embaladora, a segunda é a análise da impressora e por fim a análise do programa.

Análise da maquina empacotadora

Tendo em conta que o processo de selagem é baseado no aquecimento das mandibulas, que ao detetar o papel, ambas deslocam-se exercendo um aperto aos dois papeis, permitindo que as bordas do papel de embalagem derretam sobre as bordas do papel de etiqueta, esta foi a primeira parte a ser analisada.

Ao se acionar a maquina empacotadora, de acordo com a ordem já indicada anteriormente, os dois papeis (papel de etiqueta e papel de embalagem) foram encaminhados ao sistema, passando estes entre as mandibulas, depois pela tesoura, contudo não selando o medicamento.

Perante este problema tomou-se como possível causa, a baixa temperatura de aquecimento das mandibulas, uma que a selagem é garantida pelo processo térmico.

Depois de grandes estudos, por meio de pesquisas bibliográficas, chegou-se a conclusão de que a temperatura de selagem é de aproximadamente 120°, tendo sido tomada como referencia a informação abaixo:

Esperar a que a temperatura das mandíbulas esteja na temperatura adequada (aproximadamente 120° C). [5]

Com a temperatura de 120° C, ao se acionar a maquina empacotadora, os dois papeis foram encaminhados ao sistema, chegando as mandibulas, contudo, não passavam para a etapa de corte.

Ao se fazer uma análise se verificou que o papel de embalagem derretia sobre a mandibula e não sobre as bordas do papel de etiqueta, facto que impedia a embalagem.

Depois de grandes estudos, por meio de pesquisas bibliográficas, chegou-se a três hipóteses, de acordo com o manual de instruções da maquina embaladora VICPACK.

Não sela bem as embalagens

Pode dever-se às seguintes razões:

- Temperatura não adequada
- Sujidade e proceder à sua limpeza se for o caso.
- A utilização de papel de má qualidade pode provocar a má selagem e até danificar a impressora. Recomenda-se a utilização somente do tipo de papel especificado pelo fabricante. [5]

Análise das hipóteses:

Ao se fazer a análise da hipótese da temperatura não adequada, esta não se considerou como a principal causa pelo facto da temperatura abaixo de 120° C não ter sido suficiente para deter o papel e a partir de 120° C, o papel derreter sobre a mandíbula, não tendo nenhuma temperatura de funcionamento estável.

Ao se fazer a análise da hipótese de sujidade nas mandíbulas, foi feita a limpeza da mandíbula por meio de escovas conforme os cuidados indicados no manual de instruções da máquina embaladora VICPACK, contudo ao se iniciar o processo, verificou-se o mesmo problema, tendo sido esta também uma hipótese descartada.

A terceira hipótese implica o estudo do tipo de papel especificado pelo fabricante:

Estudo do tipo de papel

Papel Offset - Papel de impressão, com ou sem revestimento. Tem boa colagem interna e superficial, exige elevada rigidez e resistência, inclusive à água e à umidade.

Papel couché - Indicado para trabalhos de alta qualidade gráfica, como rótulos de embalagens, revistas, folhetos e encartes. É produzido, normalmente, a partir do papel de imprimir, mediante a aplicação de tinta, podendo receber acabamento brilhante ou texturizado.

Papel jornal ou papel imprensa - Destina-se à impressão de jornais, revistas, listas telefônicas, suplementos e encartes promocionais.

Papel apergaminhado - Indicado para escrever. Opaco e liso por igual na duas faces, é usado normalmente para correspondências e para produzir cadernos escolares, envelopes e folhas almaço.

Cartolina para impressos - Usado para impressos, pastas para arquivos e cartões de visita.

Papéis auto-adesivos - Recobertos por adesivo à base de resina, aderem à superfície com a qual entram em contato. Compõem etiquetas e fitas adesivas para fechar embalagens.

Papéis metalizados - Recebem revestimento metálico para fins industriais.

Papéis crepados - Por ser crepado, tem maior elasticidade e maciez, características importantes para o uso como base para germinação de sementes e fitas adesivas.

Papel de segurança - Destina-se à impressão de selos, papel-moeda, ingressos, e documentos que exigem proteção contra fraudes. Os principais recursos de segurança incluem ausência de fluorescência, microcápsulas e fios visíveis a olho nu ou sob luz UV (ultravioleta).

Para o embalamento neste equipamento foi usado um papel com revestimento termosoldável, o qual permite que o lado do revestimento derreta com o aumento da temperatura e para a etiqueta foi usado o papel couché.

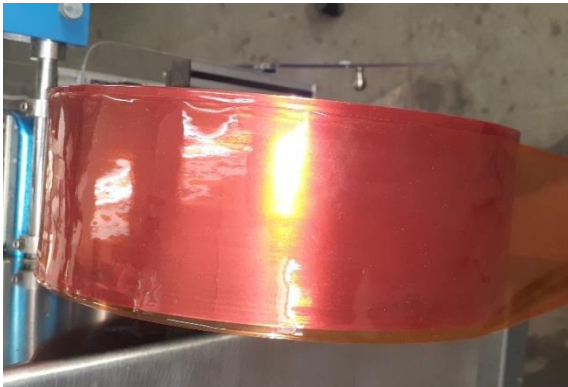


Figura 7: papel com revestimento termo-soldável para a embalagem

3.3.2. IMPRESSORA DE ETIQUETAS

A etiqueta a ser apresentada na embalagem do comprimido é gerada por meio de um programa personalizado e depois impressa por meio de uma impressora *Datamax M-Class Mark II*, sendo esta uma impressora desenvolvida especificamente para impressão de etiquetas, conforme ilustrado na figura abaixo.



Figura 8: Impressora Datamax M-Class Mark II

Conforme a figura acima ilustrada, pode-se verificar que a impressora é composta por duas partes: painel de controle e a parte da impressão.

Abaixo é ilustrado o painel de controle:



Figura 9: Painel de controle da impressora Datamax M-Class Mark II.

Cada parte que compõem o painel de controle tem a sua função específica, conforme ilustra a figura abaixo.



Figura 10: Funções específicas do painel de controle da impressora Datamax M-Class Mark II

O menu do usuário contém seleções básicas desses menus:

- *Media Settings* (configurações de mídia)
- *Print Control*
- *Printer Options* (opções de impressora)
- *System Settings* (configurações do sistema)

Tabela 10: Itens exibidos no menu do usuário

ITEM EXIBIDO	DESCRIÇÃO DO ITEM
<i>MEDIA TYPE</i>	Seleciona o método usado para imprimir etiquetas e deve ser definido de acordo com o tipo de mídia sendo usado.
<i>DIRECT THERMAL</i>	Define o uso de mídia que reage ao calor para produzir uma imagem.
<i>THERMAL TRANSFER</i>	Define o uso de mídia que requer uma fita para produzir uma imagem.

<i>SENSOR TYPE</i>	Selecione o método de detecção de topo de formulário (TOF) usado para determinar a extremidade inicial da etiqueta.
<i>GAP</i>	TOF será reconhecido detectando as aberturas ou entalhes na mídia. (Configuração padrão)
<i>CONTINUOUS</i>	Não será usada detecção de TOF; em vez disso, será usado o COMPRIMENTO DA ETIQUETA (em Configurações de mídia).
<i>REFLECTIVE</i>	O TOF será reconhecido detectando as marcas reflexivas (pretas) no lado inferior da mídia.
<i>LABEL LENGTH</i>	Determina o comprimento da etiqueta (0 - 99,99 polegadas) quando o TIPO DE SENSOR é definido para CONTÍNUO.
04.00	É a configuração padrão.
<i>MAXIMUM LABEL LENGTH</i>	Define a distância (0 - 99.99 polegadas) pela qual a impressora alimentará a mídia para encontrar o TOF (quando o Tipo de sensor for definido para ABERTURA ou REFLEXIVO) antes que uma falha de TOF seja declarada.
16.00	É a configuração padrão.

3.3.2.1. Conexão de interface

A impressora pode ser conectada ao *host* por meio de interface paralela, USB, serial ou de rede opcional. A impressora se conectará automaticamente à primeira porta que fornecer dados válidos. Depois de estabelecida, a energia da impressora deve ser ligada e desligada para mudar uma conexão de interface. [2]

A **Conexão paralela** requer um cabo Centronics IEEE 1284 com um conector macho de 36 pinos para comunicações unidirecionais (canal dianteiro), ou um cabo compatível com IEEE 1284 para comunicações bidirecionais (canais dianteiro e traseiro).

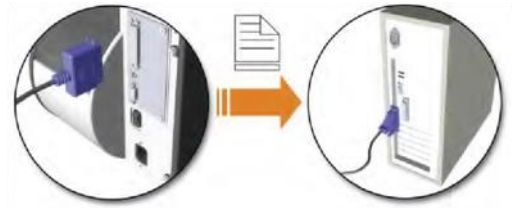


Figura 11: Conexão Paralela

A **Conexão USB** requer um cabo USB e tem suporte nos sistemas operacionais Windows 95 e mais recentes. Dependendo do sistema operacional do computador, os requisitos da instalação poderão variar ligeiramente.

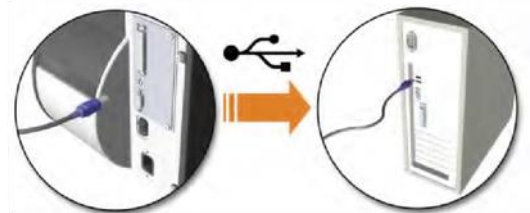


Figura 12: Conexão USB

A **Conexão serial** requer um cabo de interface serial com pinagem específica para comunicações adequadas. A interface dá suporte a comunicações RS-232C por meio de um conector DB9

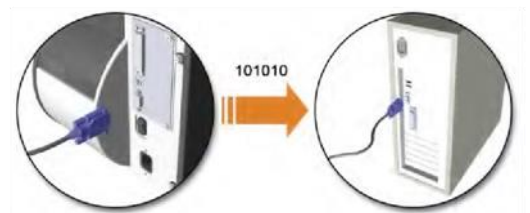


Figura 13: Conexão serial

Ao ligar a impressora, automaticamente ela entra no estado de espera, este estado permite a recepção dados à serem impressos.

Ao se realizar o teste da impressora, por meio do botão teste, nada era impresso sobre o papel, face a este problema foi feita uma pesquisa bibliográfica, por meio da qual se encontrou a forma correcta de colocação da fita de impressão conforme se ilustra na figura abaixo:

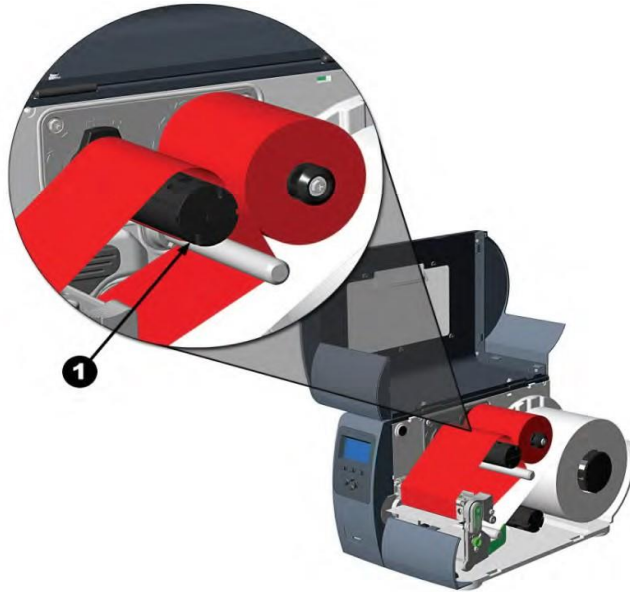


Figura 14: Encaminhamento de fita com o lado do revestimento para dentro

Tendo vencido o problema de colocação incorrecta da fita, foi-se para etapa seguinte: o teste da impressora através do envio de informação por meio do computador. Para tal foi necessário a instalação do *driver* da impressora, sendo este o elemento principal para que haja comunicação entre a impressora e o computador.

3.3.3. PROGRAMA GERADOR DE ETIQUETAS

Tendo vencido os problemas encontrados no estudo da maquina embaladora e na impressora seguiu-se para o estudo do computador, especificamente o programa gerador de etiquetas la contido. O programa originalmente usado para gerar as etiquetas a serem impressas na embalagem é o *VICPACK 14*, permite a introdução de dados tais como:

- Nome do hospital e Nome do comprimido
- Data de empacotamento e Validade;

Após a inserção dos dados, este são impressos no papel de embalagem, seguido o seguinte processo:

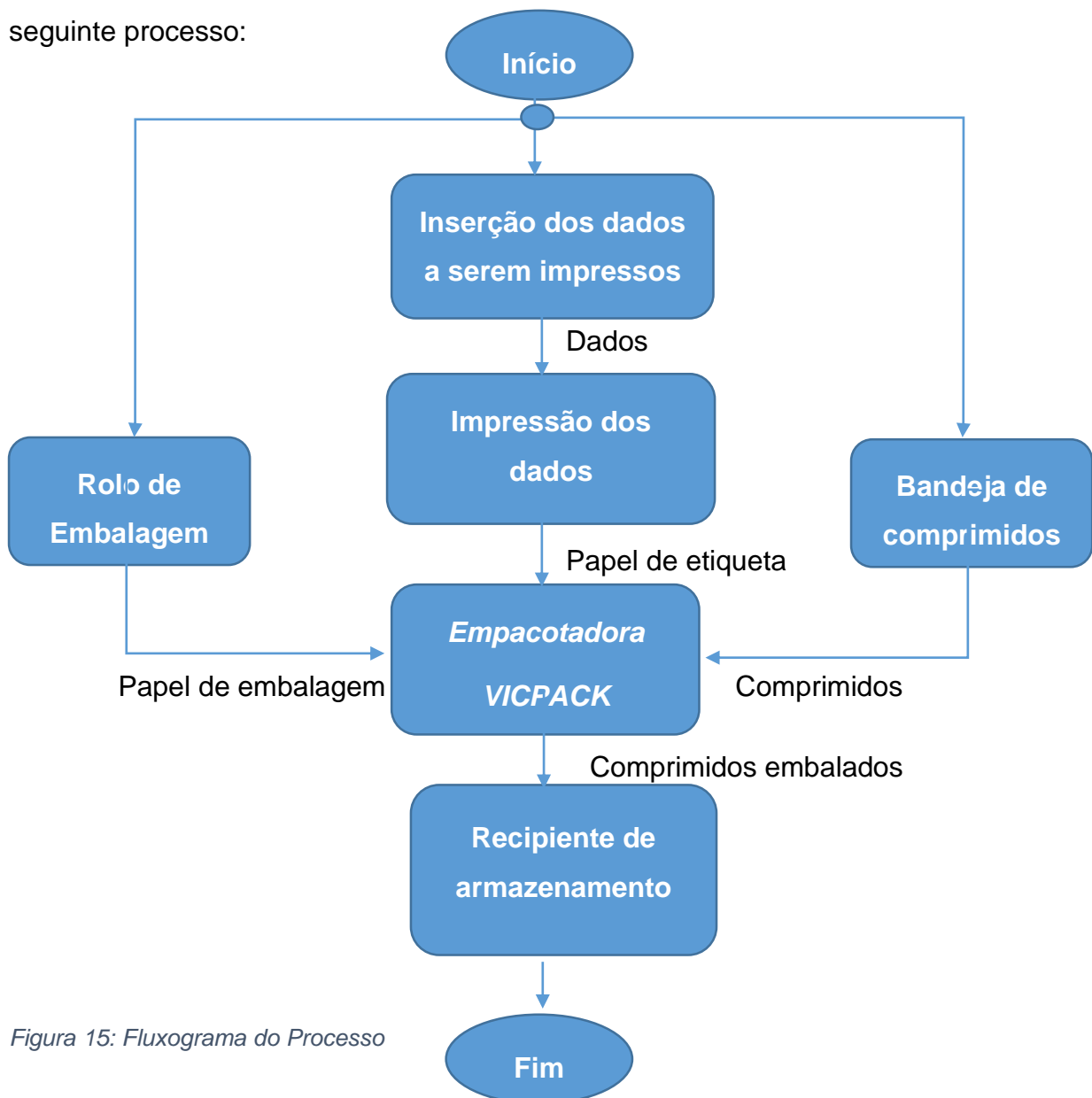


Figura 15: Fluxograma do Processo

Após a introdução dos dados no programa gerador de etiquetas, estes foram enviados à impressora por meio de uma *interface* serial, contudo a impressora não reconheceu os dados enviados por meio desta *interface*. Foram feitos testes de envio de dados em outros programas por meio da interface *USB*, os quais foram bem-sucedidos, contudo o programa gerador de etiquetas VICPACK 14 foi pré-determinado para enviar dados por meio da interface serial, facto que impossibilita o uso deste programa, desta forma surge a necessidade da criação de um novo programa gerador de etiquetas.

3.3.3.1. DESENVOLVIMENTO DO PROGRAMA GERADOR DE ETIQUETAS

Pela natureza do programa a desenvolver, foi usado a linguagem de programação java, através do *NetBeans IDE 8.2*, que contem um ambiente que permite facilidades de programação gráfica.

Um **programa** em Java é um conjunto de classes, uma das quais será a principal. Uma **classe** pode ser vista como um conjunto de variáveis e subprogramas(métodos).

Métodos são blocos de instruções que contêm uma identificação, realizam determinada tarefa e podem retornar algum tipo de informação. Método *main ()* é o método principal. Abaixo é apresentado o fluxograma do programa:

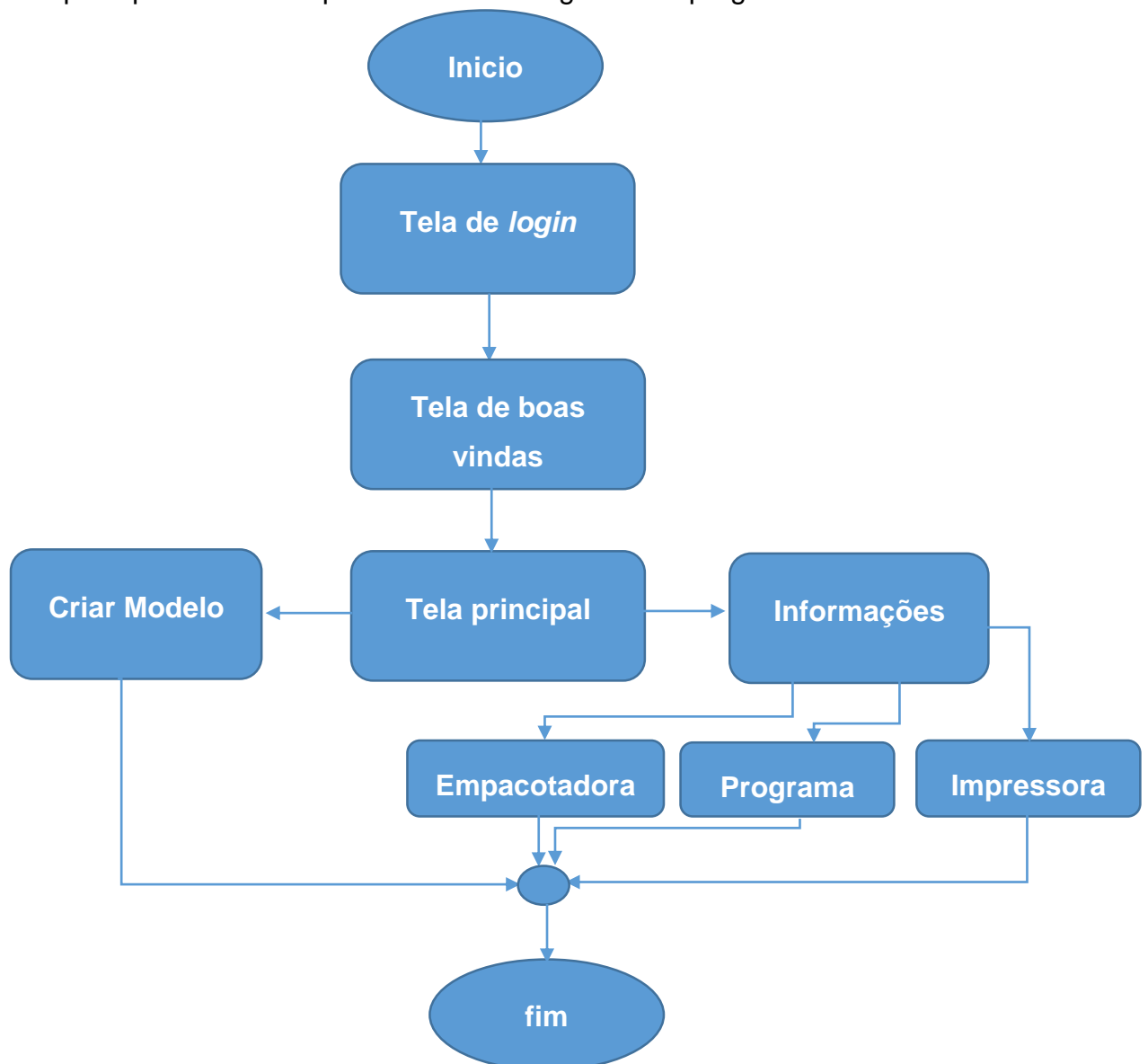


Figura 16: Fluxograma do programa

A tela de *login* permite o acesso ao sistema, foi colocada uma palavra passe para que somente pessoas autorizadas tenham o seu acesso, não foram limitadas o numero de tentativas do acesso. Abaixo é apresentada a tela de *login*:

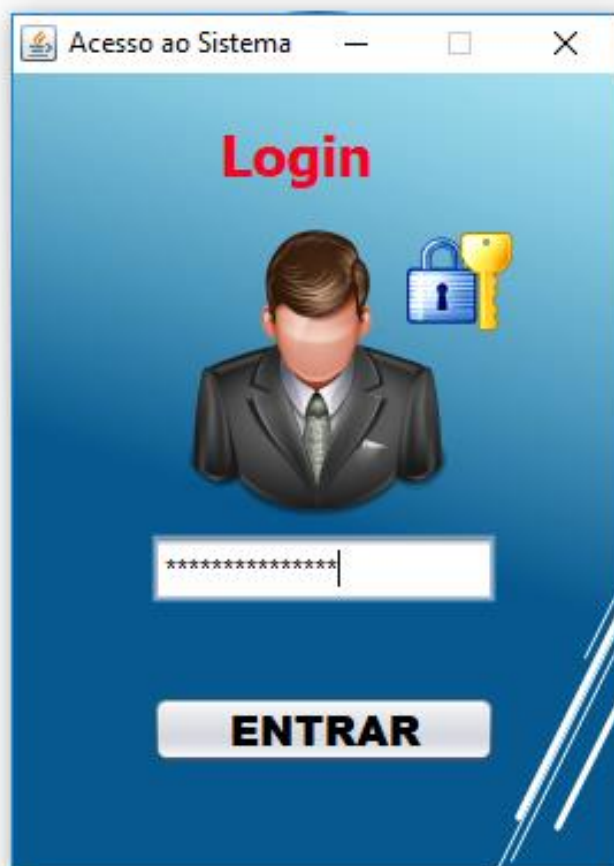


Figura 17: Tela do Login

Foi criada uma *interface* que permite a facilidade de interação com o utilizador, ao se falhar a palavra passe é demonstrada uma tela, conforme a mensagem abaixo:

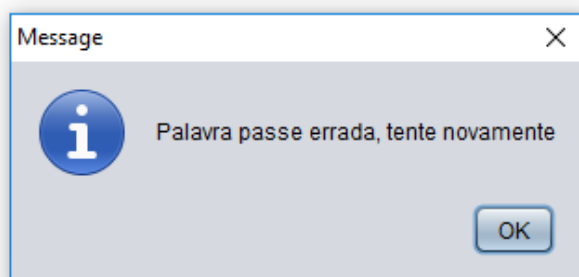


Figura 18: Mensagem de erro de senha

Após a validação da palavra passe, o utilizador recebe uma mensagem de boas vindas, conforme ilustrado abaixo:



Figura 19: Tela de boas vindas

A tela de boas vindas é composta pela mensagem de boas vindas e pelo botão avançar, o qual permite o acesso a tela principal, conforme ilustrado abaixo:

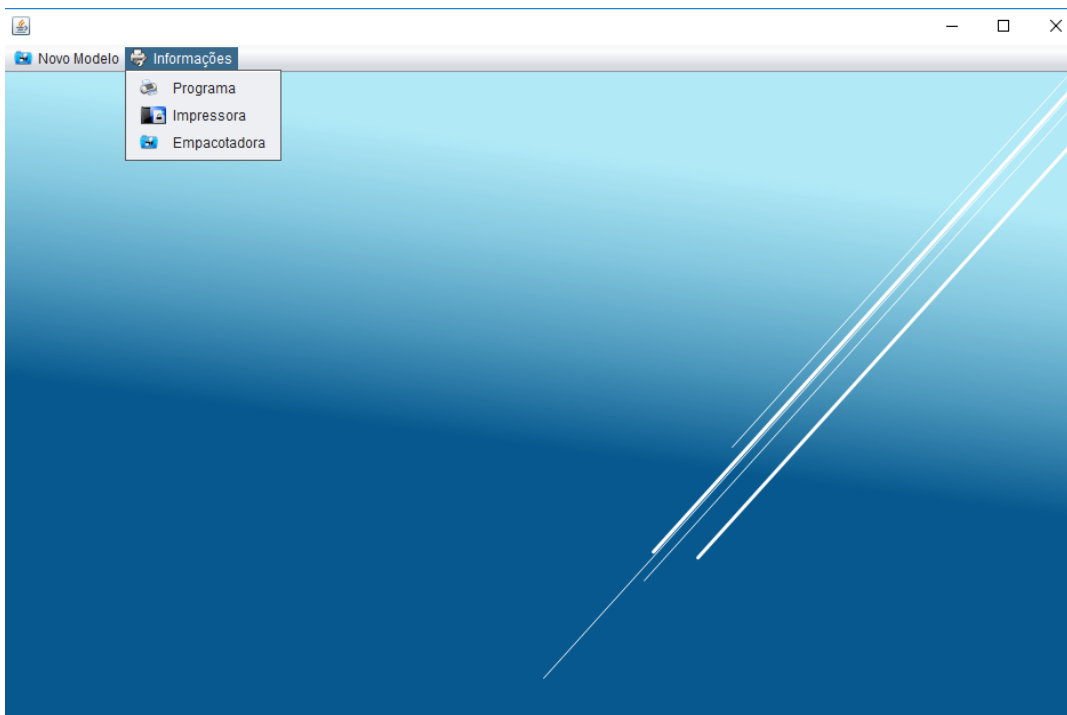


Figura 20: Tela Principal

A tela principal é onde se encontram os métodos que permitem a criação da etiqueta a ser impressa no comprimido, também foi colocada nela a tela de informação, que permite a obtenção de informação relativamente ao programa, a impressora, bem como a empacotadora.

Abaixo é apresentada a tela de criação do modelo:

Criando novo modelo...

Nome do Hospital: HCM

Nome do Comprimido: Paracetamol

Quantidade:

Data de Embalagem:

Data de Validade:

Deixar longe do alcance de crianças
Verificar a validade antes de usar

Visualizar Limpar

Imprimir Sair

Figura 21: Tela de criação do modelo

A tela de criação do modelo contém o campo de inserção dos dados, campo de informação e o campo de visualização.

Campo de inserção de dados

- Nome do Hospital;
- Nome do Comprimido;
- Quantidade;
- Data de Embalagem;
- Data de Validade;

Campo de Informação

Este campo contem uma informação que deve ser lida pelo paciente antes de fazer o uso do medicamento, nomeadamente:

- Deixar longe do alcance de crianças;
- Verificar a validade antes de usar.

Campo de visualização

O campo de visualização permite verificar a informação a ser impressa, a qual pode ser alterada se houver algum erro.

Depois de visualizada, ao clicar o botão imprimir, o programa permite realizar operações tais como: ajustar o tamanho, modificar a cor, escolher a impressora, etc.

Os dados apresentados no campo de informações podem ser vistos nos anexos, sendo estes resultados das simulações feitas, junto a estes dados é apresentado o código do programa.

O teste de funcionamento do equipamento foi feito no sector da manutenção, tendo sido obtido os resultados ilustrados abaixo:

CAPÍTULO 4

SIMULAÇÕES E TESTES

Neste capítulo são apresentados os resultados obtidos por meio de simulação e testes realizados no sector da micromecânica do SMH, por meio dos quais foram corrigidos alguns erros.

A simulação foi feita por meio do *Netbeans 8.2*, abaixo seguem os resultados obtidos na fase de simulação e teste:

4.1. PROGRAMA

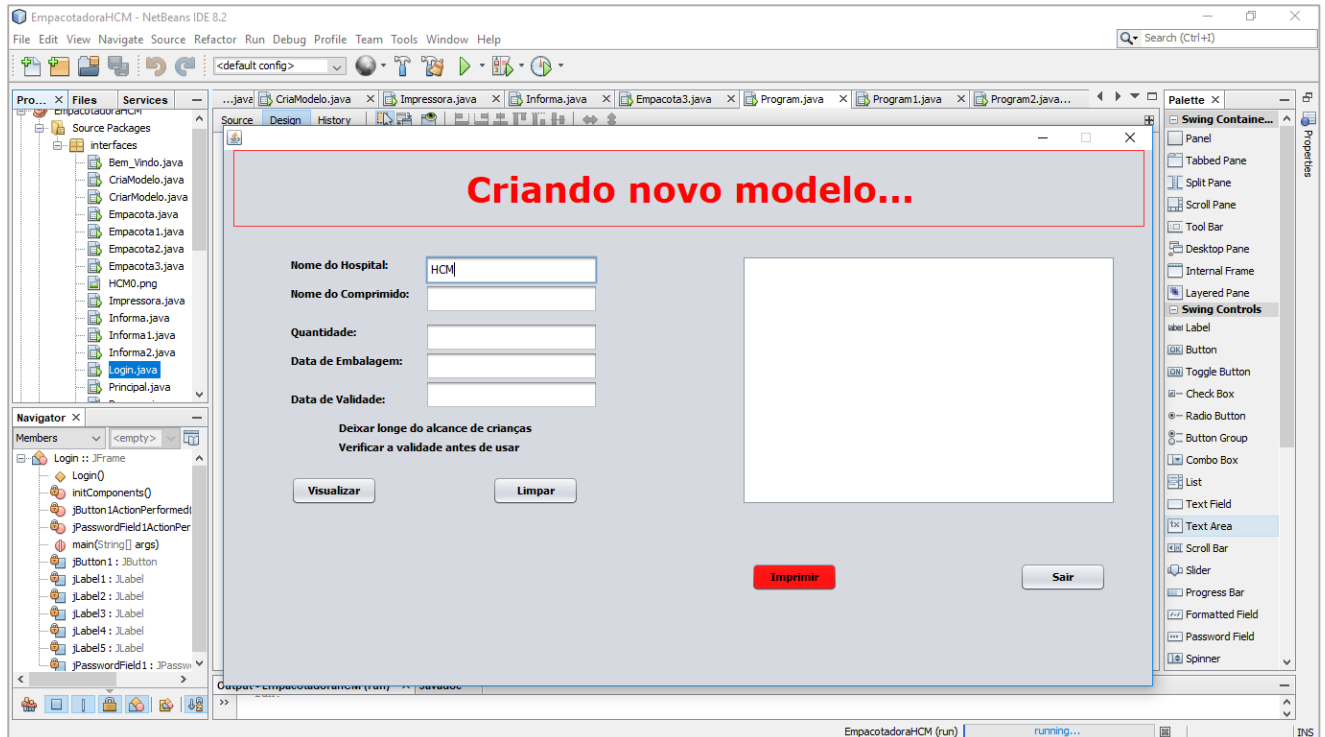


Figura 22: Teste do programa gerador de etiquetas

4.2. IMPRESSORA



Figura 23: Teste do posicionamento dos rolos na Impressora

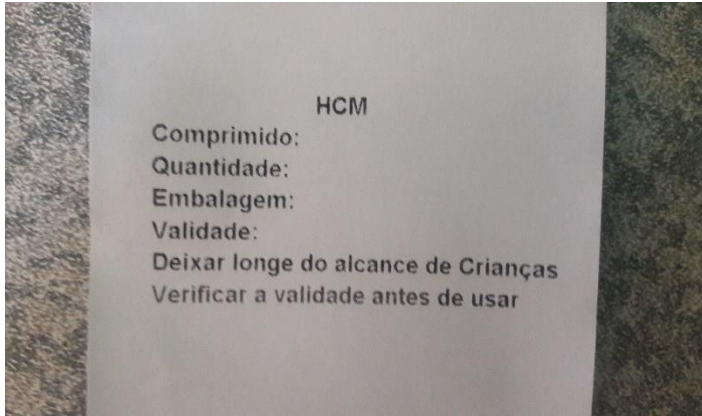


Figura 24: Teste de impressão sem visualizar os dados inseridos

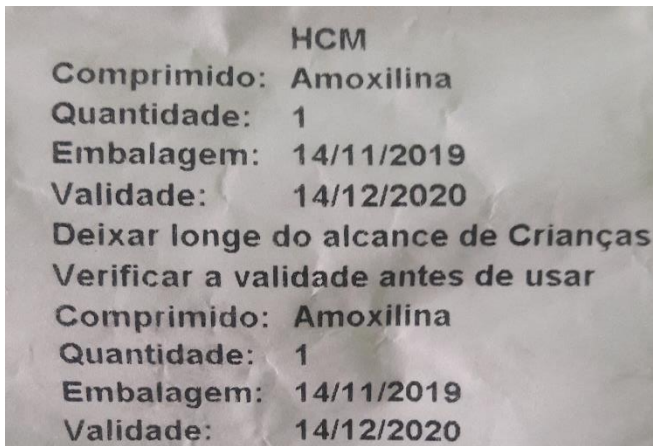


Figura 25: Teste de Impressão com múltiplas visualizações

4.3. EMPACOTADORA



Figura 26: Empacotamento com erro de corte e selagem

O erro foi corrigido com o ajuste da temperatura das mandíbulas e o ajuste da altura do corte da tesoura.



Figura 27: Ajuste da temperatura das mandíbulas



Figura 28: Ajuste da altura da tesoura



Mandibula

Tesoura

Figura 29: Processo de selagem

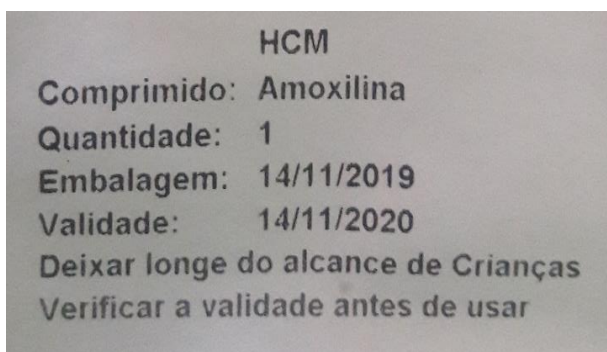


Figura 30: Teste final da etiqueta



Figura 31: Teste final do empacotamento de comprimido

CAPÍTULO 5

CONSIDERAÇÕES FINAIS

Sendo este o ultimo capitulo são apresentadas as considerações finais do trabalho, as conclusões obtidas por meio dos resultados alcançados, as recomendações e referências bibliográficas

5.1. CONCLUSÃO

Findo o trabalho, pode-se concluir que a implementação do sistema *VICPACK* no HCM é de extrema importância e urgência, pelo facto do nível da demanda de comprimidos aumentar a cada ano e o sistema de empacotamento manual se tornar cada vez mais trabalhoso e desgastante.

O estudo do modo de funcionamento da máquina empacotadora *VICPACK* foi feito por meio de uma pesquisa qualitativa, tendo sido o manual de utilização da empacotadora a principal fonte.

O *Driver* da impressora foi o principal requisito para estabelecer a comunicação entre a máquina empacotadora *VICPACK* e a impressora *datamax-o'neil*.

O programa para registar os dados a serem impressos na embalagem foi criado com sucesso, usando a linguagem de programação java através do *NetBeans IDE 8.2*;

Por meio das simulações e testes realizados, pode se concluir que o sistema pode ser implementado.

5.2. RECOMENDAÇÕES

A engenharia é um processo contínuo, por meio do qual são utilizados os recursos existentes para responder os problemas da humanidade. Muitas vezes os problemas não são resolvidos na totalidade, sendo assim são apresentadas as seguintes recomendações para o melhoramento do sistema:

- Desenvolvimento de uma base de dados onde serão armazenados os dados introduzidos na criação de novos modelos de comprimidos.
- Desenvolvimento de um sistema de detecção e correção de erros de empacotamento.

5.3. BIBLIOGRAFIA

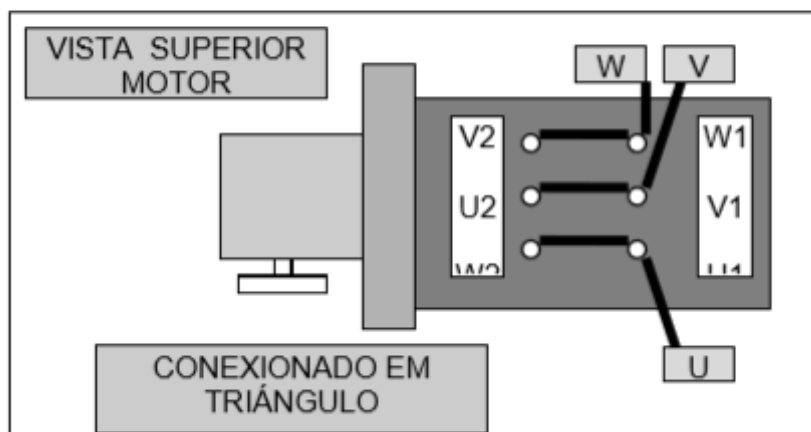
- [1] CHULALONGKORN UNIVERSITY. 2011- *Learning Computer Programming Using Java With 101 Examples*, 1ª Edição, Tailândia, acessado em <https://www.cp.eng.chula.ac.th/books/wpcontent/uploads/sites/5/2018/01/java101>
- [2] DATAMAX-O'NEIL. 2010- *Manual do Operador*, 1ª Edição, acessado em <http://www.datamax-oneil.com/>
- [3] ECK, David. 2007- *Introduction to Programming Using Java*, 5ª Edição, Califórnia.
- [4] HONEYWELL. 2018- *Software Release Notes Windows Driver 2017*, acessado em <http://www.datamax-oneil.com/>
- [5] SLIDELOG. 2018- *Manual de instruções da Máquina de reembalamento de medicamentos em dose unitária Viclina HF200A*, 1ª Edição;
- [6] <http://www.hcm.gov.mz/historia-do-hospital/> acessado em 18/11/2019 as 12:26
- [7] <http://www.hcm.gov.mz/manutencao-hospitalar/> acessado em 18 de 11 de 2019 as 12:39
- [8] <http://www.hcm.gov.mz/organizacao-administrativa/> acessado em 18/11/2019 as 12:30

ANEXOS

Especificações técnicas da Máquina Embaladora VICPACK



Manual de Instruções - Máquina Embaladora VICPACK



Esta indicação da tensão de trabalho encontra-se indicada no equipamento estes encontram-se preparados para evitar erros.

De seguida apresenta-se uma etiqueta de identificação do motor, conexão e voltagens de funcionamento:

MOTOR 3 ~ 50 / 60 Hz	IEC 34 IP 55 CL. F
TEPE MU63B . 4	COS ϕ 0.67 / 0.69
KW 0.18 / 0.22	1360 / 1650 min -1
Y 380 - 420 / 440 - 480 V	0.75 / 0.75 A
∇ 220 - 240 / 250 - 280 V	1.3 / 1.3 A

O driver do Windows está localizado no CD-ROM de acessórios incluído com a sua impressora. Para obter a versão mais recente, visite o nosso web site em www.datamax-oneil.com.

Instalação do driver do Windows:

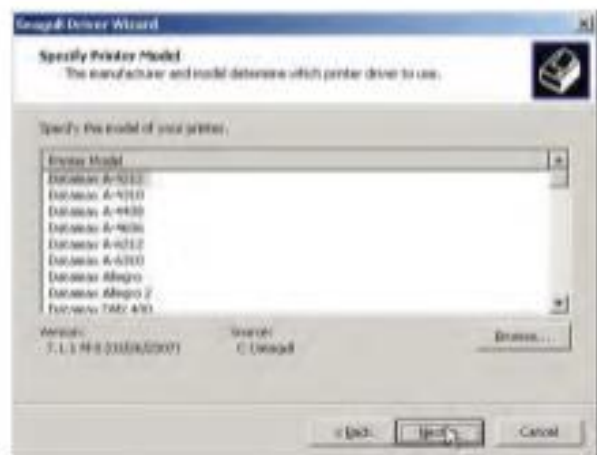
Coloque o CD-ROM de acessórios incluído com a sua impressora na unidade de CD-ROM do computador.



Quando o CD-ROM iniciar, selecione "Instalar driver do Windows" no menu principal e siga as instruções da tela para instalar.



Quando solicitado, selecione a sua impressora na lista, (ex.: Datamax-O'Neil M-Class MarkII). Continue a seguir as instruções da tela para instalar o driver.



Código gerador de etiquetas

```

Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x Informa.java
Source Design History
1  ...5 lines
6  package interfaces;
7
8  import javax.swing.JOptionPane;
9
10 /**...4 lines */
14 public class Login extends javax.swing.JFrame {
15
16     /** Creates new form Login ...3 lines */
19     public Login() {
20         initComponents();
21     }
22
23     /** This method is called from within the constructor to initialize the form ...5 lines */
28     @SuppressWarnings("unchecked")
29     // <editor-fold defaultstate="collapsed" desc="Generated Code">
30     private void initComponents() {
31
32         jLabel2 = new javax.swing.JLabel();
33         jLabel3 = new javax.swing.JLabel();
34         jButton1 = new javax.swing.JButton();
35         jPasswordField1 = new javax.swing.JPasswordField();
36         jLabel1 = new javax.swing.JLabel();
37         jLabel4 = new javax.swing.JLabel();
38         jLabel5 = new javax.swing.JLabel();
39
40         jLabel2.setToolTipText("logico");
41
42         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
43         setTitle("Acesso ao Sistema");
44         setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
45         setResizable(false);
46
47         getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());
48
49         jLabel3.setFont(new java.awt.Font("Tahoma", 1, 24)); // NOI18N
50         jLabel3.setForeground(new java.awt.Color(255, 0, 30));
51         jLabel3.setText("Login");
52         getContentPane().add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(90, 20, 80, 30));
53
54         jButton1.setFont(new java.awt.Font("Arial Black", 0, 18)); // NOI18N
55         jButton1.setText("ENTRAR");
56         jButton1.addActionListener(new java.awt.event.ActionListener() {
57             public void actionPerformed(java.awt.event.ActionEvent evt) {
58                 jButton1ActionPerformed(evt);
59             }
60         });
61         getContentPane().add(jButton1, new org.netbeans.lib.awtextra.AbsoluteConstraints(60, 270, 150, 30));
62
63         jPasswordField1.setText("jPasswordField1");
64         jPasswordField1.addActionListener(new java.awt.event.ActionListener() {
65             public void actionPerformed(java.awt.event.ActionEvent evt) {
66                 jPasswordField1ActionPerformed(evt);
67             }
68         });
69         getContentPane().add(jPasswordField1, new org.netbeans.lib.awtextra.AbsoluteConstraints(60, 200, 150, 30));
70
71         jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/interfaces/user-icon.png"))); // NOI18N
72         getContentPane().add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(70, 60, 130, 140));
73
74         jLabel4.setIcon(new javax.swing.ImageIcon("C:\\Users\\Edy Langa\\Desktop\\Edy1\\secrecy-icon (1).png")); // NOI18N
75         getContentPane().add(jLabel4, new org.netbeans.lib.awtextra.AbsoluteConstraints(170, 60, 60, 60));
76
77         jLabel5.setIcon(new javax.swing.ImageIcon("C:\\Users\\Edy Langa\\Desktop\\Edy1\\fundo_login1.png")); // NOI18N

```


RELATÓRIO DE ESTÁGIO- HCM

```
Source Design History
76 JLabel5.setIcon(new javax.swing.ImageIcon("C:\\Users\\Edu Langa\\Desktop\\Edu1\\fundo_login1.png")); // NOI18N
77 JLabel5.setText("jLabel5");
78 getContentPane().add(jLabel5, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 270, 360));
79
80 setSize(new java.awt.Dimension(269, 375));
81 setLocationRelativeTo(null);
82 } // </editor-fold>
83
84 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
85     if (jPasswordField1.getText().equalsIgnoreCase("empacotadorahcm"))
86     {new Bem_Vindo().setVisible(true);
87     dispose();
88     }
89     else {JOptionPane.showMessageDialog(null, "Palavra passe errada, tente novamente");
90     }
91 }
92 }
93 private void jPasswordField1ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }
96
97 /**...3 lines */
100 public static void main(String args[]) {...31 lines }
131
132 // Variables declaration - do not modify
133 private javax.swing.JButton jButton1;
134 private javax.swing.JLabel jLabel1;
135 private javax.swing.JLabel jLabel2;
136 private javax.swing.JLabel jLabel3;
137 private javax.swing.JLabel jLabel4;
138 private javax.swing.JLabel jLabel5;
139 private javax.swing.JPasswordField jPasswordField1;
140 // End of variables declaration
141 }
```

```
Source Design History
76 JLabel5.setIcon(new javax.swing.ImageIcon("C:\\Users\\Edu Langa\\Desktop\\Edu1\\fundo_login1.png")); // NOI18N
77 JLabel5.setText("jLabel5");
78 getContentPane().add(jLabel5, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 270, 360));
79
80 setSize(new java.awt.Dimension(269, 375));
81 setLocationRelativeTo(null);
82 } // </editor-fold>
83
84 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
85     if (jPasswordField1.getText().equalsIgnoreCase("empacotadorahcm"))
86     {new Bem_Vindo().setVisible(true);
87     dispose();
88     }
89     else {JOptionPane.showMessageDialog(null, "Palavra passe errada, tente novamente");
90     }
91 }
92 }
93 private void jPasswordField1ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }
96
97 /**...3 lines */
100 public static void main(String args[]) {...31 lines }
131
132 // Variables declaration - do not modify
133 private javax.swing.JButton jButton1;
134 private javax.swing.JLabel jLabel1;
135 private javax.swing.JLabel jLabel2;
136 private javax.swing.JLabel jLabel3;
137 private javax.swing.JLabel jLabel4;
138 private javax.swing.JLabel jLabel5;
139 private javax.swing.JPasswordField jPasswordField1;
140 // End of variables declaration
141 }
```

RELATÓRIO DE ESTÁGIO- HCM

```
Source Design History
6 package interfaces;
7
8 /**...4 lines */
12 public class Bem_Vindo extends javax.swing.JFrame {
13
14     /** Creates new form Bem_Vindo ...3 lines */
17     public Bem_Vindo() {
18         initComponents();
19     }
20
21     /** This method is called from within the constructor to initialize the form ...5 lines */
26     @SuppressWarnings("unchecked")
27     // <editor-fold defaultstate="collapsed" desc="Generated Code">
28     private void initComponents() {
29
30         jLabel3 = new javax.swing.JLabel();
31         jButton1 = new javax.swing.JButton();
32         jLabel2 = new javax.swing.JLabel();
33         jLabel1 = new javax.swing.JLabel();
34
35         setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
36         setResizable(false);
37         getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());
38
39         jLabel3.setFont(new java.awt.Font("Verdana", 1, 14)); // NOI18N
40         jLabel3.setForeground(new java.awt.Color(255, 255, 255));
41         jLabel3.setText("Avançar");
42         getContentPane().add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(830, 390, 70, 30));
43
44         jButton1.setIcon(new javax.swing.ImageIcon("C:\\Users\\Edy Langa\\Desktop\\Edy1\\avancar.png")); // NOI18N
45         jButton1.setMnemonic('A');
46         jButton1.setAutoscrolls(true);
47         jButton1.addActionListener(new java.awt.event.ActionListener() {
```

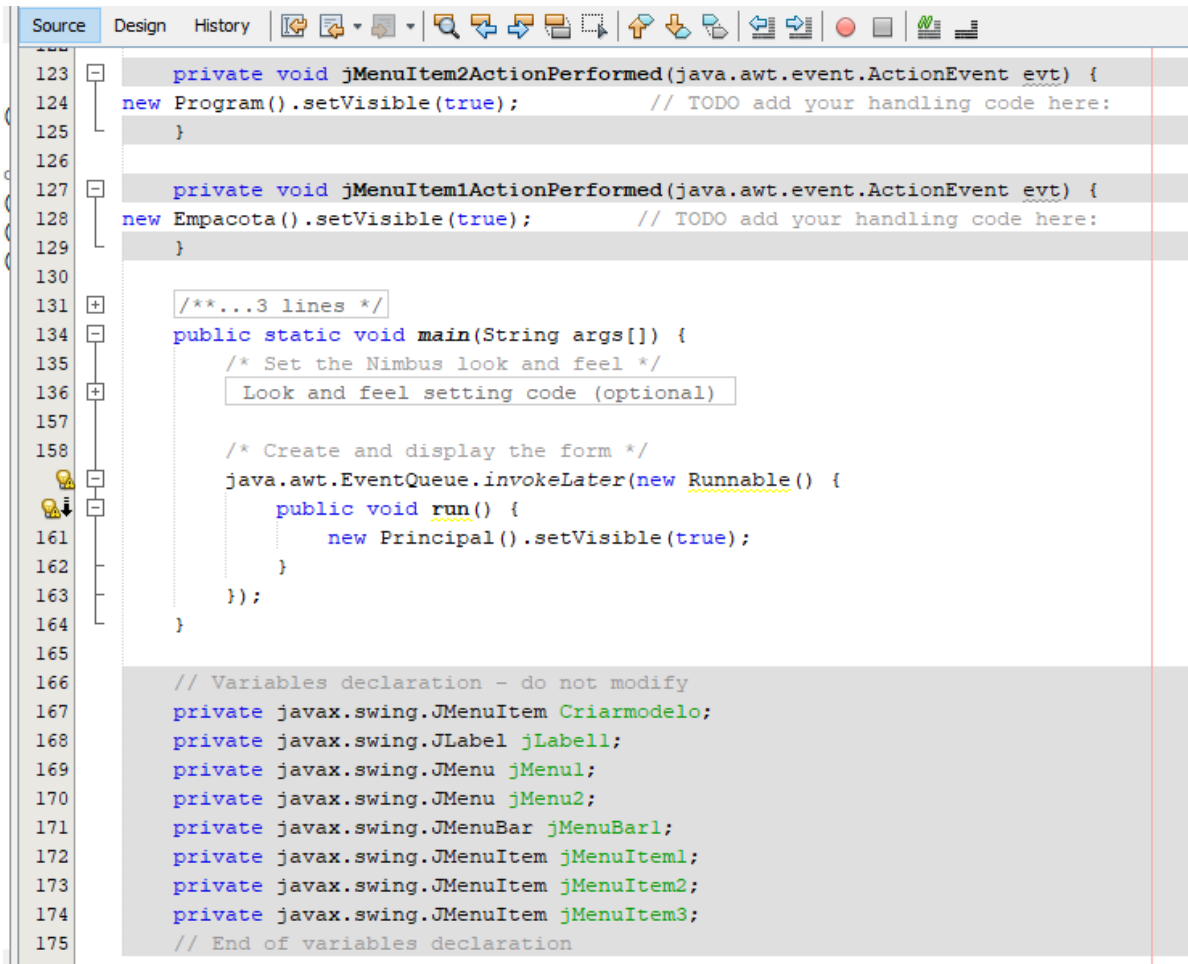
```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x Informa.java x Empacota3.java x Program.java...
Source Design History
49     public void actionPerformed(java.awt.event.ActionEvent evt) {
50         jButton1ActionPerformed(evt);
51     }
52     getContentPane().add(jButton1, new org.netbeans.lib.awtextra.AbsoluteConstraints(810, 420, 110, 100));
53
54     jLabel2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/interfaces/HCM0.png"))); // NOI18N
55     jLabel2.setText("jLabel2");
56     getContentPane().add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(390, 80, 170, 140));
57
58     jLabel1.setDisplayedMnemonic('m');
59     jLabel1.setForeground(new java.awt.Color(255, 0, 0));
60     jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/interfaces/fundo_project1.png"))); // NOI18N
61     getContentPane().add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, -150, 950, 870));
62
63     setSize(new java.awt.Dimension(950, 571));
64     setLocationRelativeTo(null);
65 } // </editor-fold>
66
67 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
68     new Principal().setVisible(true);
69     dispose();
70 }
71
72
73 /**...3 lines */
76 public static void main(String args[]) { ...31 lines }
107
108 // Variables declaration - do not modify
109 private javax.swing.JButton jButton1;
110 private javax.swing.JLabel jLabel1;
111 private javax.swing.JLabel jLabel2;
112 private javax.swing.JLabel jLabel3;
```

RELATÓRIO DE ESTÁGIO- HCM

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x
Source Design History
62
63     setSize(new java.awt.Dimension(950, 571));
64     setLocationRelativeTo(null);
65 } // </editor-fold>
66
67 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
68     new Principal().setVisible(true);
69     dispose();
70
71 }
72
73 /**...3 lines */
76 public static void main(String args[]) { ...31 lines }
107
108 // Variables declaration - do not modify
109 private javax.swing.JButton jButton1;
110 private javax.swing.JLabel jLabel1;
111 private javax.swing.JLabel jLabel2;
112 private javax.swing.JLabel jLabel3;
113 // End of variables declaration
114 }
115
```

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x Informa.java x Em
Source Design History
1 ...5 lines
6 package interfaces;
7
8 /**...4 lines */
12 public class Principal extends javax.swing.JFrame {
13
14     /** Creates new form Principal ...3 lines */
17     public Principal() {
18         initComponents();
19     }
20
21     /** This method is called from within the constructor to initialize the form ...5 lines */
26     @SuppressWarnings("unchecked")
27     Generated Code
110
111     private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt) {
112     new Impressora().setVisible(true); // TODO add your handling code here:
113     }
114
115     private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {
116     }
117
118
119     private void CriarModeloActionPerformed(java.awt.event.ActionEvent evt) {
120     new CriarModelo().setVisible(true);
121     }
122
123     private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {
124     new Program().setVisible(true); // TODO add your handling code here:
125     }
126
127     private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {
```

RELATÓRIO DE ESTÁGIO- HCM



```
123 private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {
124     new Program().setVisible(true);          // TODO add your handling code here:
125 }
126
127 private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {
128     new Empacota().setVisible(true);        // TODO add your handling code here:
129 }
130
131 /**...3 lines */
132
133 public static void main(String args[]) {
134     /* Set the Nimbus look and feel */
135     Look and feel setting code (optional)
136
137     /* Create and display the form */
138     java.awt.EventQueue.invokeLater(new Runnable() {
139         public void run() {
140             new Principal().setVisible(true);
141         }
142     });
143 }
144
145 // Variables declaration - do not modify
146 private javax.swing.JMenuItem Criarmodelo;
147 private javax.swing.JLabel jLabel1;
148 private javax.swing.JMenu jMenu1;
149 private javax.swing.JMenu jMenu2;
150 private javax.swing.JMenuBar jMenuBar1;
151 private javax.swing.JMenuItem jMenuItem1;
152 private javax.swing.JMenuItem jMenuItem2;
153 private javax.swing.JMenuItem jMenuItem3;
154 // End of variables declaration
```

RELATÓRIO DE ESTÁGIO- HCM

```
1 ...5 lines
6 package interfaces;
7
8 import java.awt.print.PrinterException;
9 import java.util.logging.Level;
10 import java.util.logging.Logger;
11 import javax.swing.JFrame;
12 import javax.swing.JOptionPane;
13
14 /**...4 lines */
18 public class CriarModelo extends javax.swing.JFrame {
19
20     /** Creates new form CriarModelo ...3 lines */
23     public CriarModelo() {
24         initComponents();
25     }
26
27     /** This method is called from within the constructor to initialize the form
32     @SuppressWarnings("unchecked")
33     // <editor-fold defaultstate="collapsed" desc="Generated Code">
34     private void initComponents() {
35
36         jPanel1 = new javax.swing.JPanel();
37         jLabel1 = new javax.swing.JLabel();
38         jScrollPane1 = new javax.swing.JScrollPane();
39         jTextArea1 = new javax.swing.JTextArea();
40         jLabelUso = new javax.swing.JLabel();
41         jLabelHospital = new javax.swing.JLabel();
42         jLabelComprimido = new javax.swing.JLabel();
43         jLabelQuantidade = new javax.swing.JLabel();
44         jLabelEmbalagem = new javax.swing.JLabel();
45         jLabelValidade = new javax.swing.JLabel();
```

```
45         jLabelValidade = new javax.swing.JLabel();
46         jLabelCrianças = new javax.swing.JLabel();
47         hospital = new javax.swing.JTextField();
48         nomeComp = new javax.swing.JTextField();
49         quant = new javax.swing.JTextField();
50         embalagem = new javax.swing.JTextField();
51         validade = new javax.swing.JTextField();
52         jButtonLimpar = new javax.swing.JButton();
53         jButtonVisualizar = new javax.swing.JButton();
54         jButtonImprimir = new javax.swing.JButton();
55         jButtonSair = new javax.swing.JButton();
56
57         setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
58         setBackground(new java.awt.Color(0, 102, 102));
59         setResizable(false);
60         getContentPane().setLayout(null);
61
62         jPanel1.setBorder(javax.swing.BorderFactory.createMatteBorder(1, 1, 1, 1, new java.awt.Color(255, 51, 51)));
63         jPanel1.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N
64
65         jLabel1.setFont(new java.awt.Font("Verdana", 1, 36)); // NOI18N
66         jLabel1.setForeground(new java.awt.Color(255, 3, 3));
67         jLabel1.setText("Criando novo modelo...");
68
69         javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
70         jPanel1.setLayout(jPanel1Layout);
71         jPanel1Layout.setHorizontalGroup(
72             jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
73                 .addGroup(jPanel1Layout.createSequentialGroup()
74                     .addGap(242, 242, 242)
75                     .addComponent(jLabel1)
76                     .addGap(239, Short.MAX_VALUE))
```

RELATÓRIO DE ESTÁGIO- HCM

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriModelo.java x Impressora.java x Informa.java x Empacota:
Source Design History
76         .addContainerGap(239, Short.MAX_VALUE)
77     );
78     jPanel1Layout.setVerticalGroup(
79         jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
80         .addGroup(jPanel1Layout.createSequentialGroup()
81             .addContainerGap()
82             .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, 56, Short.MAX_VALUE)
83             .addContainerGap()
84         );
85
86     getContentPane().add(jPanel1);
87     jPanel1.setBounds(10, 0, 950, 80);
88
89     jTextArea1.setColumns(20);
90     jTextArea1.setFont(new java.awt.Font("Arial", 1, 10)); // NOI18N
91     jTextArea1.setRows(5);
92     jScrollPane1.setViewportView(jTextArea1);
93
94     getContentPane().add(jScrollPane1);
95     jScrollPane1.setBounds(540, 110, 390, 260);
96
97     jLabelUso.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
98     jLabelUso.setText("Verificar a validade antes de usar");
99     getContentPane().add(jLabelUso);
100    jLabelUso.setBounds(120, 300, 210, 20);
101
102    jLabelHospital.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
103    jLabelHospital.setText("Nome do Hospital:");
104    getContentPane().add(jLabelHospital);
105    jLabelHospital.setBounds(70, 110, 110, 20);
106
107    jLabelComprimido.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
```

RELATÓRIO DE ESTÁGIO- HCM

```
Source Design History
107      jLabelComprimido.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
108      jLabelComprimido.setText("Nome do Comprimido:");
109      getContentPane().add(jLabelComprimido);
110      jLabelComprimido.setBounds(70, 140, 130, 20);
111
112      jLabelQuantidade.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
113      jLabelQuantidade.setText("Quantidade:");
114      getContentPane().add(jLabelQuantidade);
115      jLabelQuantidade.setBounds(70, 180, 110, 20);
116
117      jLabelEmbalagem.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
118      jLabelEmbalagem.setText("Data de Embalagem:");
119      getContentPane().add(jLabelEmbalagem);
120      jLabelEmbalagem.setBounds(70, 210, 150, 20);
121
122      jLabelValidade.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
123      jLabelValidade.setText("Data de Validade:");
124      getContentPane().add(jLabelValidade);
125      jLabelValidade.setBounds(70, 250, 110, 20);
126
127      jLabelCrianças.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
128      jLabelCrianças.setText("Deixar longe do alcance de crianças");
129      getContentPane().add(jLabelCrianças);
130      jLabelCrianças.setBounds(120, 280, 210, 20);
131
132      hospital.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
133      hospital.setText("HCM");
134      hospital.addActionListener(new java.awt.event.ActionListener() {
135          public void actionPerformed(java.awt.event.ActionEvent evt) {
136              hospitalActionPerformed(evt);
137          }
138      });
```

```
Source Design History
138      });
139      getContentPane().add(hospital);
140      hospital.setBounds(210, 110, 180, 30);
141
142      nomeComp.addActionListener(new java.awt.event.ActionListener() {
143          public void actionPerformed(java.awt.event.ActionEvent evt) {
144              nomeCompActionPerformed(evt);
145          }
146      });
147      getContentPane().add(nomeComp);
148      nomeComp.setBounds(210, 140, 180, 30);
149
150      quant.addActionListener(new java.awt.event.ActionListener() {
151          public void actionPerformed(java.awt.event.ActionEvent evt) {
152              quantActionPerformed(evt);
153          }
154      });
155      getContentPane().add(quant);
156      quant.setBounds(210, 180, 180, 30);
157
158      embalagem.addActionListener(new java.awt.event.ActionListener() {
159          public void actionPerformed(java.awt.event.ActionEvent evt) {
160              embalagemActionPerformed(evt);
161          }
162      });
163      getContentPane().add(embalagem);
164      embalagem.setBounds(210, 210, 180, 30);
165
166      validade.addActionListener(new java.awt.event.ActionListener() {
167          public void actionPerformed(java.awt.event.ActionEvent evt) {
168              validadeActionPerformed(evt);
```


RELATÓRIO DE ESTÁGIO- HCM

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x In
Source Design History
168         validadeActionPerformed(evt);
169     }
170 }
171 getContentPane().add(validade);
172 validade.setBounds(210, 240, 180, 30);
173
174 jButtonLimpar.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
175 jButtonLimpar.setText("Limpar");
176 jButtonLimpar.addActionListener(new java.awt.event.ActionListener() {
177     public void actionPerformed(java.awt.event.ActionEvent evt) {
178         jButtonLimparActionPerformed(evt);
179     }
180 });
181 getContentPane().add(jButtonLimpar);
182 jButtonLimpar.setBounds(280, 340, 90, 30);
183
184 jButtonVisualizar.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
185 jButtonVisualizar.setText("Visualizar");
186 jButtonVisualizar.addActionListener(new java.awt.event.ActionListener() {
187     public void actionPerformed(java.awt.event.ActionEvent evt) {
188         jButtonVisualizarActionPerformed(evt);
189     }
190 });
191 getContentPane().add(jButtonVisualizar);
192 jButtonVisualizar.setBounds(70, 340, 90, 30);
193
194 jButtonImprimir.setBackground(new java.awt.Color(255, 0, 0));
195 jButtonImprimir.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
196 jButtonImprimir.setText("Imprimir");
197 jButtonImprimir.addActionListener(new java.awt.event.ActionListener() {
198     public void actionPerformed(java.awt.event.ActionEvent evt) {
199         jButtonImprimirActionPerformed(evt);
```

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x In
Source Design History
199         jButtonImprimirActionPerformed(evt);
200     }
201 }
202 getContentPane().add(jButtonImprimir);
203 jButtonImprimir.setBounds(550, 430, 90, 30);
204
205 jButtonSair.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N
206 jButtonSair.setText("Sair");
207 jButtonSair.addActionListener(new java.awt.event.ActionListener() {
208     public void actionPerformed(java.awt.event.ActionEvent evt) {
209         jButtonSairActionPerformed(evt);
210     }
211 });
212 getContentPane().add(jButtonSair);
213 jButtonSair.setBounds(830, 430, 90, 30);
214
215 setSize(new java.awt.Dimension(974, 587));
216 setLocationRelativeTo(null);
217 } // </editor-fold>
218
219 private void hospitalActionPerformed(java.awt.event.ActionEvent evt) {
220     // TODO add your handling code here:
221 }
222
223 private void nomeCompActionPerformed(java.awt.event.ActionEvent evt) {
224     // TODO add your handling code here:
225 }
226
227 private void quantActionPerformed(java.awt.event.ActionEvent evt) {
228     // TODO add your handling code here:
229 }
```


RELATÓRIO DE ESTÁGIO- HCM

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x Inform
Source Design History
229 }
230
231 private void embalagemActionPerformed(java.awt.event.ActionEvent evt) {
232     // TODO add your handling code here:
233 }
234
235 private void validadeActionPerformed(java.awt.event.ActionEvent evt) {
236     // TODO add your handling code here:
237 }
238
239 private void jButtonLimparActionPerformed(java.awt.event.ActionEvent evt) {
240     JTextAreal.setText("");
241     hospital.setText("");
242     nomeComp.setText("");
243     quant.setText("");
244     embalagem.setText("");
245     validade.setText("");
246
247     // TODO add your handling code here:
248 }
249
250 private void jButtonVisualizarActionPerformed(java.awt.event.ActionEvent evt) {
251     JTextAreal.append("\t" +hospital.getText()+
252         "\n Comprimido: \t"+nomeComp.getText()+
253         "\n Quantidade: \t"+ quant.getText()+
254         "\n Embalagem: \t"+embalagem.getText()+
255         "\n Validade: \t"+ validade.getText()+
256         "\n Deixar longe do alcance de Crianças " +
257         "\n Verificar a validade antes de usar");
258     // TODO add your handling code here:
259 }
```

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x Informa.java x Empacota3.java x Program.ja
Source Design History
259 }
260
261 private void jButtonImprimirActionPerformed(java.awt.event.ActionEvent evt) {
262     try {
263         JTextAreal.print(); // TODO add your handling code here:
264     } catch (PrinterException ex) {
265         Logger.getLogger(CriarModelo.class.getName()).log(Level.SEVERE, null, ex);
266     }
267 }
268 private JFrame frame;
269 private void jButtonSairActionPerformed(java.awt.event.ActionEvent evt) {
270     frame= new JFrame();
271     if (JOptionPane.showConfirmDialog (frame," Deseja realmente sair? ",
272         " Sistema de Empacotamento",JOptionPane.YES_NO_OPTION)==JOptionPane.YES_NO_OPTION)
273     {System.exit(0);}
274
275     // TODO add your handling code here:
276 }
277
278 /**
279  * @param args the command line arguments
280  */
281 public static void main(String args[]) {
282
283     try {
284         for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
285             if ("Nimbus".equals(info.getName())) {
286                 javax.swing.UIManager.setLookAndFeel(info.getClassName());
287                 break;
288             }
289         }
290     } catch (ClassNotFoundException ex) {
```

RELATÓRIO DE ESTÁGIO- HCM

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x Informa.java x Empacota3.java x Program.java...
Source Design History
} catch (ClassNotFoundException ex) {
291     java.util.logging.Logger.getLogger(CriarModelo.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
292 } catch (InstantiationException ex) {
293     java.util.logging.Logger.getLogger(CriarModelo.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
294 } catch (IllegalAccessException ex) {
295     java.util.logging.Logger.getLogger(CriarModelo.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
296 } catch (javax.swing.UnsupportedLookAndFeelException ex) {
297     java.util.logging.Logger.getLogger(CriarModelo.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
298 }
299 //</editor-fold>
300
301 /* Create and display the form */
302 java.awt.EventQueue.invokeLater(new Runnable() {
303     public void run() {
304         new CriarModelo().setVisible(true);
305     }
306 });
307
308
309 // Variables declaration - do not modify
310 private javax.swing.JTextField embalagem;
311 private javax.swing.JTextField hospital;
312 private javax.swing.JButton jButtonImprimir;
313 private javax.swing.JButton jButtonLimpar;
314 private javax.swing.JButton jButtonSair;
315 private javax.swing.JButton jButtonVisualizar;
316 private javax.swing.JLabel jLabel1;
317 private javax.swing.JLabel jLabelComprimido;
318 private javax.swing.JLabel jLabelCrianças;
319 private javax.swing.JLabel jLabelEmbalagem;
320 private javax.swing.JLabel jLabelHospital;
```

```
Bem_Vindo.java x Principal.java x Login.java x CriarModelo.java x CriarModelo.java x Impressora.java x
Source Design History
java.awt.EventQueue.invokeLater(new Runnable() {
304     public void run() {
305         new CriarModelo().setVisible(true);
306     }
307 });
308
309 // Variables declaration - do not modify
310 private javax.swing.JTextField embalagem;
311 private javax.swing.JTextField hospital;
312 private javax.swing.JButton jButtonImprimir;
313 private javax.swing.JButton jButtonLimpar;
314 private javax.swing.JButton jButtonSair;
315 private javax.swing.JButton jButtonVisualizar;
316 private javax.swing.JLabel jLabel1;
317 private javax.swing.JLabel jLabelComprimido;
318 private javax.swing.JLabel jLabelCrianças;
319 private javax.swing.JLabel jLabelEmbalagem;
320 private javax.swing.JLabel jLabelHospital;
321 private javax.swing.JLabel jLabelQuantidade;
322 private javax.swing.JLabel jLabelUso;
323 private javax.swing.JLabel jLabelValidade;
324 private javax.swing.JPanel jPanel1;
325 private javax.swing.JScrollPane jScrollPane1;
326 private javax.swing.JTextArea jTextArea1;
327 private javax.swing.JTextField nomeComp;
328 private javax.swing.JTextField quant;
329 private javax.swing.JTextField validade;
330 // End of variables declaration
331
332 }
```