

UNIVERSIDADE LUTERANA DO BRASIL
CURSO DE SISTEMAS DE INFORMAÇÃO
CÂMPUS CARAZINHO



SISTEMA DE INVENTÁRIO DE
MATÉRIA-PRIMA

Leandro de Oliveira

Relatório de atividades desenvolvidas durante o Estágio Supervisionado e apresentadas ao Curso de Sistemas de Informação da Universidade Luterana do Brasil, campus Carazinho, como pré-requisito para a obtenção do título de Bacharel em Sistemas de Informação.

Supervisor: Daniel Di Domênico

Carazinho, Julho de 2006.

CIP – Catalogação na Publicação

De Oliveira, Leandro

SESISTEMA DE INVENTÁRIO DE MATÈRIA-PRIMA/
Leandro de Oliveira; [orientado por] Daniel Di Domênico. –
Carazinho: 2006.

29 p.: il.

Relatório de Estágio Supervisionado (Graduação em Sistemas de
Informação). Universidade Luterana do Brasil, 2006.

1. Desenvolvimento. 2. Software. 3. Gerenciamento. I. Di
Dômenico, Daniel. II. Sistema de inventário de matéria-prima.

Endereço:
Universidade Luterana do Brasil – Câmpus Carazinho
BR 285 KM 336
CEP 99500-000 Carazinho - RS - Brasil

SUMÁRIO

LISTA DE FIGURAS	5
LISTA DE ABREVIATURAS E SIGLAS	6
RESUMO	7
ABSTRACT	8
1 INTRODUÇÃO	9
2 TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO	10
2.1 ORACLE DATABASE 10G EXPRESS EDITION	10
2.1.1 Robusto e escalável.....	11
2.1.2 Acesso remoto permitido	11
2.1.3 Programação PL/SQL	11
2.1.4 Triggers e Stored procedures	11
2.1.5 Banco de dados transacional.....	12
2.1.6 Disponibilidade.....	12
2.2 JAVA	12
2.3 JDBC (JAVA DATABASE CONNECTIVITY)	14
3 DESENVOLVIMENTO DO SISTEMA	16
3.1 AMBIENTE DE DESENVOLVIMENTO	16
3.2 IMPLEMENTAÇÃO.....	16
3.2.1 Criação de funções PL/SQL	17
3.2.2 Implementação de métodos	17
3.2.3 Elaboração de interface das telas	18
4 FUNCIONAMENTO DO SISTEMA	20
5 CONCLUSÃO	28
REFERÊNCIAS	29

LISTA DE FIGURAS

Figura 1 – Arquitetura java.....	13
Figura 2 – Portabilidade java.....	14
Figura 3 – Arquitetura JDBC	15
Figura 4 – Exemplo função PL/SQL	17
Figura 5 – Método Conecta	18
Figura 6 – Manutenção de cadastro de matérias-prima.....	19
Figura 7 – Tela de autenticação de usuário	20
Figura 8 – Tela principal	21
Figura 9 – Manutenção de embalagens	22
Figura 10 – Incluído embalagem.....	22
Figura 11 – Selecionando item do cadastro.....	23
Figura 12 – Tela de edição de dados	24
Figura 13 – Excluindo registro selecionado	24
Figura 13 – Filtrando itens da tabela	25
Figura 14 – Lançamento do inventário 01.....	26
Figura 15 – Lançamento do Inventário 02.....	26
Figura 16 – Lançamento do inventario 03.....	27
Figura 17 – Lançamento do inventário 04.....	27

LISTA DE ABREVIATURAS E SIGLAS

IDE	Integrated Development Environment
PL/SQL	Procedural Language/Structured Query Language
SQL	Structured Query Language
TCP/IP	Transmission Protocol / Internet Protocol
HTTP	Hyper Text Transfer Protocol
FTP	File Transfer Protocol
JVM	Java Virtual Machine
JDBC	Java Database Connectivity
API	Application Program Interface
SGDB	Sistema Gerenciador de Banco de Dados

RESUMO

O **Sistema de inventário de matéria-prima** é um software de gerenciamento desenvolvido para a empresa Mekal Produtos Químicos Ltda., cujo principal objetivo é a automação das tarefas realizadas pela mesma no setor de produção. O sistema armazena informações sobre as matérias-primas utilizadas para produção de tintas industriais, além de possibilitar aos operadores de produção uma maior praticidade, segurança e produtividade em seu trabalho. O respectivo sistema foi desenvolvido utilizando a linguagem de programação Java, o ambiente de desenvolvimento NetBeans IDE 5.0 e tendo como banco de dados Oracle Database 10g Express Edition.

Palavras-chaves: Desenvolvimento; Software; Gerenciamento.

ABSTRACT

Title: *“RAW MATERIAL INVENTORY SYSTEM”*

The raw material inventory system is software of management developed for the company Mecal Produtos Químicos Ltda., whose main objective is the automation of the task carried out by the same one in the sector of output. The system stores information about the raw materials utilized for output of industrial ink, beyond enable the users of output a bigger facility, security and productivity in his work. The respective system was developed utilizing the programming language Java, the environment of development NetBeans IDE 5.0 and having the as database Oracle Database 10g Express Edition

Key-words: *Development; Software; Management.*

1 INTRODUÇÃO

Atualmente, empresas de todos os portes necessitam cada vez mais de auxílio em seu gerenciamento e organização administrativa. Para auxiliar essas empresas existem vários tipos de softwares que abrangem várias áreas, oferecendo recursos que possam tornar mais eficaz o gerenciamento e a execução dos serviços realizados pelas empresas. Por conta da competitividade comercial existente no mercado entre as empresas desenvolvedoras de softwares, está se tornando cada vez mais acessíveis à aquisição de softwares para diferentes tipos de empresa. Um software é de extrema importância para uma empresa, pois ajuda na diminuição de custos, aumento de produção e lucro, dentre outros fatores importantes.

A empresa Mekal Produtos Químicos Ltda., localizada na cidade de Carazinho-RS dedica-se a atender a região central do estado, atuando suas atividades nos segmentos de fabricação e distribuição de tintas industriais.

Tendo em vista o ramo que a Mekal opera e o crescimento na área de produção que obteve nos últimos anos, obrigou a empresa a buscar uma alternativa para redução de custos e obter mais agilidade nas operações de gerenciamento de matérias-prima de produção. Mediante a esse fato, notou-se a necessidade de um software que faça este controle.

Este estágio se propõe a automatizar as tarefas relativas ao controle das matérias-primas de produção da Mekal através do desenvolvimento de um **Sistema de inventário de matéria-prima**. Seu objetivo é facilitar o gerenciamento da produção, agilizar processos, reduzir custos e retornar com rapidez informações úteis para o desenrolar do processo de produção da empresa.

2 TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO

Um dos pontos fundamentais na concepção e implementação de um sistema, sem dúvida são as tecnologias que serão empregadas, com elas temos a possibilidade de acrescentar mudanças aos meios, adquirindo resultados adicionais para o desenvolvimento.

Neste capítulo vamos abordar as principais tecnologias necessárias para a elaboração deste estágio, detalhando suas características e o porquê da escolha das mesmas para este desenvolvimento.

2.1 ORACLE DATABASE 10G EXPRESS EDITION

O Oracle Database 10g Express Edition (Oracle Database XE), trata-se de uma versão gratuita da conhecida base de dados *Oracle 10g*.

Esta versão tem como particularidade de oferecer liberdade para desenvolver e implementar aplicativos de muitas plataformas podendo ser distribuída pré-instalada nas soluções de produtos e aplicativos sem custo algum. Além disso, traz suporte para uma grande variedade de ambientes de desenvolvimento.

O Oracle XE traz recursos de desempenho, confiabilidade e segurança da versão 2 do Banco de Dados *Oracle 10g*, pois foi construído sobre a mesma base de código. Por ser compatível com toda a família de Banco de Dados Oracle, permite aos usuários a facilidade de começar com uma solução básica, mas com toda funcionalidade exigida nas corporações (MAGALHÃES, 2006).

As próximas seções irão apresentar algumas características que foram imprescindíveis para a escolha do Banco de Dados Oracle para este desenvolvimento.

2.1.1 Robusto e escalável

A principal vantagem de se trabalhar com essa base de dados, é sem dúvida a grande capacidade de processamento e armazenamento que ela pode proporcionar. Deste modo é possível utilizar esta solução desde o início de um projeto, até quando em produção ele vier a atingir um tamanho considerável de dados. Ou seja, aplicações que hoje eventualmente sejam relativamente pequenas, poderão crescer mantendo toda a infra-estrutura já desenvolvida em sistemas e modelos de dados.

2.1.2 Acesso remoto permitido

Atualmente uma grande quantidade de empresas, em determinado momento, quando possui uma infra-estrutura propícia, visa uma rápida migração de seus sistemas “Desktop” para o ambiente *Web*. Como a recodificação de aplicações para ambiente *Web* pode ser bastante trabalhosa, a alternativa rápida para fazer esta migração e manter as aplicações Desktop em operação, seria compartilhando a base de dados no provedor.

Desta forma a aplicação pode ganhar uma versão on-line muito rapidamente, enquanto os desenvolvedores trabalham com uma nova versão do sistema, para ambiente *Web* real.

2.1.3 Programação PL/SQL

PL/SQL (*Procedural Language/Structured Query Language*) é uma extensão da linguagem padrão SQL (*Structured Query Language*) adicionando construções encontradas em linguagens procedurais, resultando em uma linguagem estrutural mais poderosa do que o SQL.

2.1.4 Triggers e Stored procedures

Uma característica que pode simplificar o desenvolvimento das aplicações é a utilizações de Triggers e procedimentos armazenados (Stored procedures), diretamente na base de dados.

Com isto ganha-se em padronização das regras de negócio, as quais independem de erros de programação da aplicação cliente, serão garantidas pelas Triggers já definidas na base de dados no momento de sua concepção, evitando duplicidade de códigos que poderiam se traduzir em erros no processamento.

As procedures por sua vez buscam transferir para dentro da base de dados rotinas que basicamente necessitem processar grandes quantidades de dados armazenados, sem a necessidade de transferir os dados entre a aplicação e o banco de dados, ganhando-se muito em performance (JEPSON, 1999).

2.1.5 Banco de dados transacional

Outra vantagem refere-se a possibilidade de se efetuar operações de forma transacional, ou seja, uma operação somente será completada se todas as operações do conjunto forem processadas com sucesso, evitando que os dados fiquem em estado inconsistente.

2.1.6 Disponibilidade

O Banco de Dados Oracle XE está disponível para Windows de 32 bits e uma grande variedade de sistemas operacionais Linux, incluindo Mandriva Linux 2006 Power Pack+, SUSE Linux Enterprise Server 9 e SUSE Linux 10 da Novell, Red Hat Enterprise Linux 4, Fedora e Ubuntu. O software pode ser baixado gratuitamente na Oracle Technology Network em: <http://www.oracle.com/technology/xs>.

2.2 JAVA

Java é uma linguagem de programação orientada a objetos, desenvolvida por uma pequena equipe de pessoas na *Sun Microsystems*. Inicialmente elaborada para ser a linguagem-base de projetos de software para produtos eletrônicos (NEWMAN, 1997).

De acordo com Cornell e Horstmann (2004), Java é toda uma plataforma, com uma biblioteca imensa, contendo grandes quantidades de códigos reaproveitáveis e, é também um ambiente de execução que fornece serviços como segurança, portabilidade para outros sistemas operacionais.

Java é uma linguagem de programação orientada a objetos, onde o projeto orientado a objetos é uma técnica de programação que se concentra nos dados e nas interfaces para esse objeto. A orientação a objetos vem provando o seu valor durante os últimos 30 anos, e é inadmissível que nos dias de hoje uma linguagem de programação não utilize os benefícios da orientação a objetos.

Java é uma linguagem distribuída e possui uma extensa biblioteca de rotinas para lidar com protocolos TCP/IP, como HTTP e FTP. Java foi elaborada para a produção de programas confiáveis em vários sentidos, além de possuir um alto nível de segurança, pois permite a construção de sistemas livres de vírus e intrusões, ou seja, *applets* carregado remotamente de outras máquinas não acessam os arquivos locais. Foi eliminado o uso de ponteiros, e possui também um coletor de lixo. Quando um objeto não possuir mais referências apontando para ele, seus dados não podem mais ser usados e a memória deve ser liberada, o coletor de lixo deverá liberar a memória na primeira oportunidade.

Java é uma linguagem compilada e interpretada, diferentemente das outras linguagens de programação. O compilador transforma seu programa em uma linguagem intermediária chamada *bytecode*, que são independentes de plataforma, e isso torna Java independente em relação à arquitetura. Mais tarde, esses *bytecodes* são interpretados através de um interpretador Java. Esse interpretador é o *Java Virtual Machine (JVM)*. Os *bytecodes* gerados podem ser interpretados em qualquer plataforma desde que haja a presença do JVM. Nas figuras 1 e 2 são mostradas a arquitetura e a portabilidade da linguagem java.

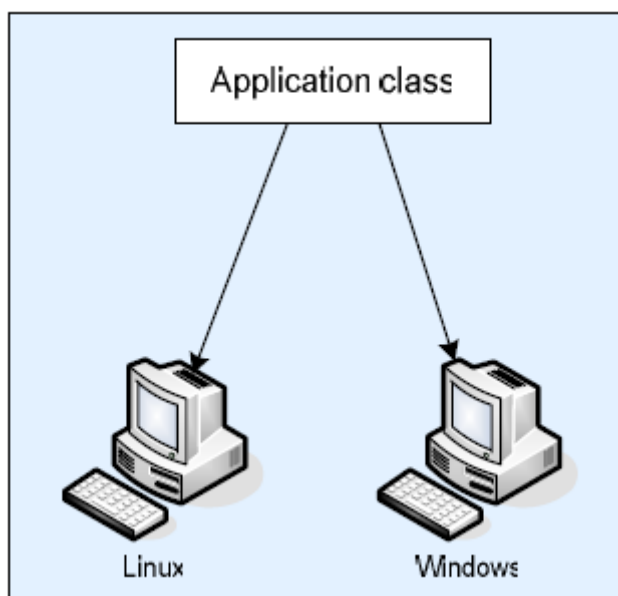


Figura 1 – Arquitetura java

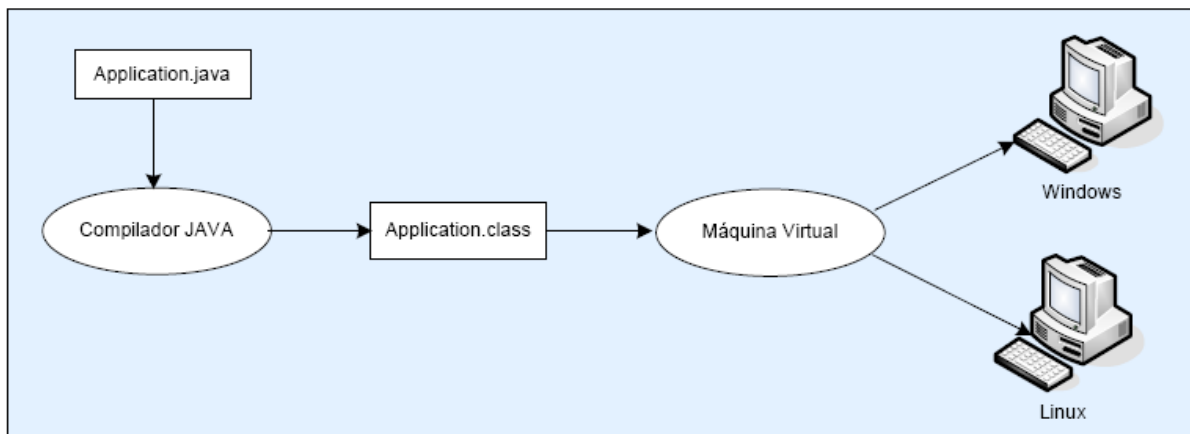


Figura 2 – Portabilidade java

Java é uma linguagem robusta, pois põe bastante ênfase na verificação rápida de possíveis problemas. Java faz uma ampla verificação em tempos de compilação e execução. Através de tratamento de exceções é possível eliminar situações propensas a erros.

2.3 JDBC (JAVA DATABASE CONNECTIVITY)

Segundo Jepson (1999), JDBC é uma API para acesso a SGDB (Sistemas Gerenciadores de Banco de Dados) por meios de comandos SQL.

Ele foi desenvolvido pela *Sun Microsystems* trabalhando em conjunto com líderes mundiais no setor de banco de dados. A proposta do JDBC é a de ser a interface do programador Java com o banco de dados, ou seja, permite que se escrevam cláusulas SQL incorporadas ao código da aplicação, possibilitando o tráfego entre o banco de dados e a aplicação Java. (JEPSON, 2005).

Para usar o JDBC, é necessário apenas carregar o *driver* JDBC para que haja a comunicação com o SGDB, sem a necessidade de desenvolver uma aplicação voltada para um banco de dados específico, pois a API para programação é a mesma para qualquer SGDB. Assim como Java, o JDBC é independente de arquitetura e plataforma.

A figura abaixo representa a arquitetura JDBC:

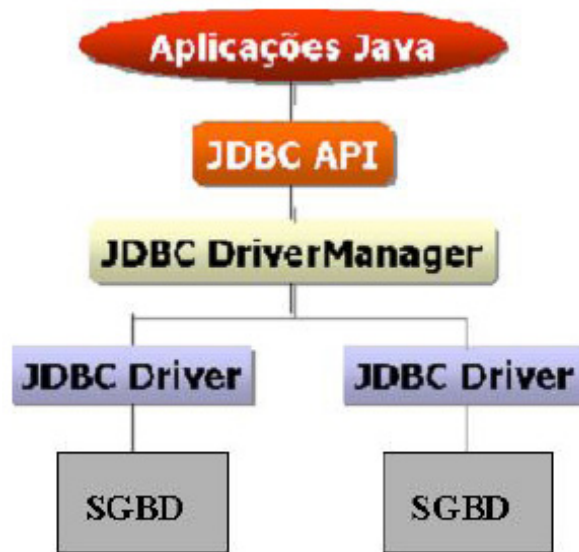


Figura 3 – Arquitetura JDBC

3 DESENVOLVIMENTO DO SISTEMA

3.1 AMBIENTE DE DESENVOLVIMENTO

Para o desenvolvimento do seguinte trabalho foram escolhidas ferramentas que atendessem a determinados requisitos, como:

- Ser software livre ou freeware;
- Ter portabilidade, para que possa ser utilizado em várias plataformas;
- Ter confiabilidade e segurança; e
- Suporte acoplamento com um banco de dados;

Desta forma, o ambiente de desenvolvimento das técnicas para este trabalho constitui do seguinte:

- Ambiente de programação: NetBeans IDE 5.0;
- Banco de dados: Oracle 10g Express Edition;

3.2 IMPLEMENTAÇÃO

Nesta seção serão apresentados alguns dos artefatos produzidos durante o desenvolvimento do sistema.

Para que haja uma maior compreensão do, esta fase foi dividida em três partes:

- Criação de funções PL/SQL
- Implementação de métodos
- Elaboração de interface das telas

3.2.1 Criação de funções PL/SQL

A grande vantagem da criação de blocos PL/SQL é que podem ser definidas todas as regras de negócio no próprio banco de dados. Proporcionando desempenho e robustez, assim como um ótimo meio de integração com outras aplicações.

Na Figura 4 é mostrada uma função PL/SQL que foi criada no desenvolvimento deste trabalho. Esta função será utilizada para retornar o valor total em quilos de uma matéria-prima específica, através de cálculos que realizam o somatório dos lançamentos em litros e faz a conversão para quilos.

```
create or replace FUNCTION "TOTAL_BAL"
(codbal in NUMBER,
aberto_lts NUMBER default 0,
aberto_kg NUMBER default 0)
return NUMBER
is
  lts_emb NUMBER(15,4);
  lts_tan NUMBER(15,4);
  total NUMBER(15,4);
  peso_e NUMBER(15,4);
BEGIN
  SELECT
    SUM(ie.total_lts) + (aberto_lts)
  INTO lts_emb
  FROM itens_embalagem ie
  WHERE ie.cod_bal = codbal;
  SELECT
    SUM((it.total_cm / ct.cm_lt))
  INTO lts_tan
  FROM itens_tanque it, cad_tanque ct
  WHERE it.cod_tanque = ct.codigo
  AND it.cod_bal = codbal;
  SELECT
    cm.peso_esp
  INTO peso_e
  FROM cad_mp cm, cad_balanco cb
  WHERE cb.cod_mp = cm.codigo
  AND cb.codigo = codbal;
  total := (((lts_emb + lts_tan) * peso_e) + aberto_kg);
  RETURN(total);
END;
```

Figura 4 – Exemplo função PL/SQL

3.2.2 Implementação de métodos

Na Figura 5 é mostrada um dos métodos implementados. O método “Conecta” tem como finalidade efetuar a conexão com o banco de dados, para isso ele efetua a leitura do arquivo “Bancos.imp”, onde estão contidas todos os parâmetros necessários para estabelecer a

conexão, contando com utilização do método instanciado “decriptografa” que tem a função de decriptografar as informações que estão cifradas.

```
public static void Conecta(){
    try {
        Security d = new Security();

        File arquivo = new File("Bancos.imp");
        FileReader le = new FileReader(arquivo);
        LineNumberReader linha = new LineNumberReader(le);

        while (linha.readLine() != null){
            if (linha.getLineNumber()==2){//Driver
                Driver = linha.readLine();
            }
            if (linha.getLineNumber()==3){//Url
                Url = linha.readLine();
            }
            if (linha.getLineNumber()==4){//Usuario
                Usuario = d.decriptografa(linha.readLine());
            }
            if (linha.getLineNumber()==5){//Senha
                Senha = d.decriptografa(linha.readLine());
            }
        }

        linha.close();
        le.close();

        Class.forName(Driver);
        conexao = DriverManager.getConnection(Url,Usuario,Senha);
    } catch(Exception ex){
        JOptionPane.showMessageDialog(null,ex);
    }
}
```

Figura 5 – Método Conecta

3.2.3 Elaboração de interface das telas

Ao definir uma interface de telas, estaremos tendo a responsabilidade de demonstrar através de ícones e menus todo o conteúdo do software, todas as suas funcionalidades, devendo assim otimizar todos os recursos disponíveis do sistema.

No momento da elaboração do layout devemos ter todo o cuidado em atrair o usuário para a aplicação, e mantê-lo na aplicação, pois qualquer ação repentina ou dificuldade que o usuário tenha pode fazer com que ele desista de utilizá-lo ou tenha que se submeter a um tempo maior em treinamentos.

Para uma maior interação entre o usuário e o sistema as telas foram elaboradas de forma simples, mas que atendessem as funcionalidades projetadas para o sistema, proporcionando mais agilidade ao operador.

Dessa maneira foi adotado um mesmo padrão para a elaboração de todas as telas de manutenção de cadastros, que pode ser observada como exemplo através das Figuras 5. Na mesma tela é possível visualizar todos os dados cadastrados, podendo a qualquer momento incluir, alterar ou excluir dados, através da ação de clique aos botões com mesma identificação, além do recurso de consulta através de filtro com parâmetros escolhidos pelo usuário.

The screenshot shows a software window titled "IMP - MATÉRIAS-PRIMA" with a sub-header "Matérias-prima". It features two radio buttons for filtering: "Código" (selected) and "Mat. Prima". Below these is a search input field. A table displays 13 rows of data with the following columns: "Código", "Matéria Prima", "Peso Esp.", and "Cm/Lt". At the bottom, there are three buttons: "Incluir" (with a plus icon), "Alterar" (with a pencil icon), and "Excluir" (with a minus icon).

Código	Matéria Prima	Peso Esp.	Cm/Lt
1	A41	1,28	16,2
2	6933730	1,09	2,72
3	A100	0,97	13,72
4	A161	0,97	39,18
5	A207	0,95	29,02
6	A214	0,97	43,55
7	A222	0,94	43,92
8	A607	0,98	40,23
9	AB043	0,86	8,77
10	AD043	0,87	27,15
11	ADT041	0,9	14,66
12	A042	1,24	8,51
13	ADT217	0,88	17,98

Figura 6 – Manutenção de cadastro de matérias-prima

4 FUNCIONAMENTO DO SISTEMA

Este capítulo demonstra a aplicação em funcionamento, descrevendo as funções realizadas pelo sistema, bem como a interface de telas desenvolvida.

De acordo com as especificações definidas pelo cliente, para ter acesso ao sistema, é necessário que o operador identifique-se com seu usuário e senha. Na Figura 7 é apresentada a tela de autenticação de usuário.



Figura 7 – Tela de autenticação de usuário

Após ter sido identificado, o sistema verifica o tipo de usuário do sistema para atribuir suas permissões de acesso. Depois de sua confirmação o usuário entra na tela principal do sistema que é apresentado na Figura 8.



Figura 8 – Tela principal

A partir da tela principal o usuário que estiver ativo poderá acessar os menus e ícones disponíveis para realizar as operações do sistema, como cadastros, consultas, lançamentos, relatórios e configurações.

Nas Figuras 9 e 10 são apresentadas as telas que compõem a manutenção de dados das embalagens.

IMP - EMBALAGENS

:: Embalagens

Código:
 Descrição:

Código	Descrição	Volume
1	EMBALAGEM_01	3,6
2	EMBALAGEM_04	0,9
3	EMBALAGEM_51	18
4	EMBALAGEM_60	198
5	EMBALAGEM_3E	4,5
6	EMBALAGEM_1F	3,3
7	EMBALAGEM_1D	3
8	EMBALAGEM_66	5
9	EMBALAGEM_77	2
11	TAMBOR RECUPERADO 198 LITROS	198

Figura 9 – Manutenção de embalagens

IMP - CADASTRO DE EMBALAGENS

:: Embalagens

Código:
Descrição:
Volume (Lts):

Figura 10 – Incluído embalagem

Através da seleção da tabela de itens cadastrados é possível realizar determinadas tarefas como alteração e exclusão, mostrada na Figura 11.



Figura 11 – Selecionando item do cadastro

Com a ação de duplo clique sobre o item que se deseja alterar, é aberta a tela de edição onde é possível alterar os dados do cadastro finalizando a operação clicando no botão salvar, conforme a Figura 12.

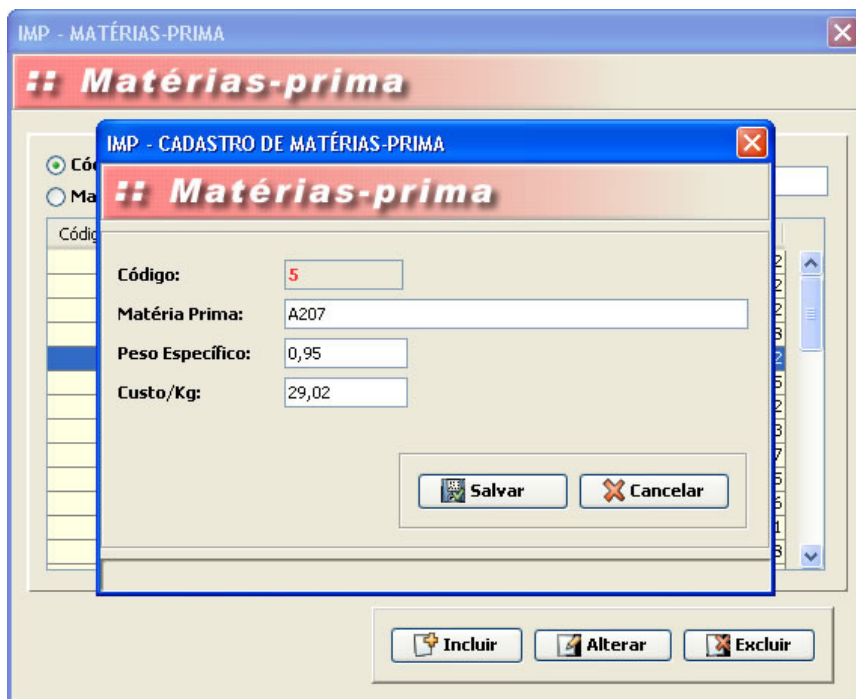


Figura 12 – Tela de edição de dados

Outra opção é a exclusão de registros, selecionando determinado item e após clicar no botão excluir. Conforme a Figura 13.

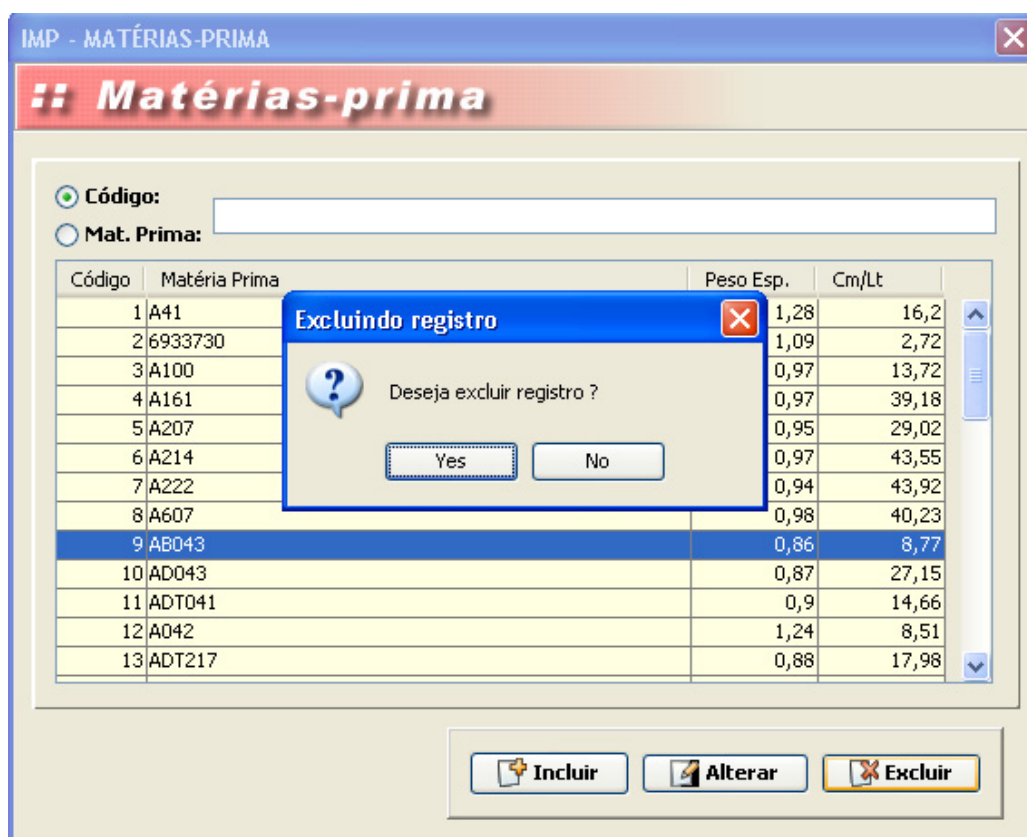


Figura 13 – Excluindo registro selecionado

Através de entrada de parâmetros pelo usuário é possível realizar uma filtragem de itens contida na tabela de registros, com a opção de escolha de tipos de parâmetros para o filtro como a descrição do item por exemplo, demonstrada na Figura 13.

Código	Descrição	Altura	Cm/Lt
3	TANQUE_3,9	162	3,9

Figura 13 – Filtrando itens da tabela

Nas Figuras 14,15,16 e 17 é possível visualizar as telas do lançamento do inventário

Este formulário concentra as informações para que possa ser gerado um resultado final de todos os itens que foram lançados. Através da escolha de determina matéria-prima, são lançadas as quantidade em litros de embalagens e dos tanques em estoque, também são consideradas quantidades que já estão abertas em fase de produção. O resultado final gera a quantidade do produto em quilos e o seu custo correspondente.

IMP - LANÇAMENTO DO INVENTÁRIO

:: Inventário

Código: **Data:** **Total (Kg):** 66,626
Matéria Prima: **Total (R\$):** 533,01

1 - Embalagens 2 - Tanques

Embalagem: **Qtde.:** **Total:**

Seq.	Código	Embalagem	Qtde.	Total (Lts)
1	2	EMBALAGEM_045	2	1,8

Qtde. Aberto (Lts) **Qtde. Aberto (Kg)**

Figura 14 – Lançamento do inventário 01

IMP - LANÇAMENTO DO INVENTÁRIO

:: Inventário

Código: **Data:** **Total (Kg):** 66,626
Matéria Prima: **Total (R\$):** 533,01

1 - Embalagens 2 - Tanques

Tanques: **Sobra (Cm):** **Total:**

Seq.	Código	Tanque	Sobra (Cm)	Total (Cm)
1	3	TANQUE_3,9	10	152

Qtde. Aberto (Lts) **Qtde. Aberto (Kg)**

Figura 15 – Lançamento do Inventário 02

IMP - LANÇAMENTO DO INVENTÁRIO

:: Inventário

Código: 3 **Data:** 30/06/2006 **Total (Kg):** 66,626

Matéria Prima: 41 D401 - R\$: 8 - Peso: 1,59 **Total (R\$):** 533,01

[F2] - Consulta Matérias-prima

1 - Embalagens 2 - Tanques

Embalagem: _____ **Qtde.:** _____ **Total:** _____

Seq.	Código	Embalagem	Qtde.	Total (Lts)
1	2	EMBALAGEM_045	2	1,8

Qtde. Aberto (Lts) 0,5 **Qtde. Aberto (Kg)** 1

Figura 16 – Lançamento do inventario 03

IMP - LANÇAMENTO DO INVENTÁRIO

:: Inventário

Código: 3 **Data:** 30/06/2006 **Total (Kg):** 66,626

Matéria Prima: 41 D401 - R\$: 8 - Peso: 1,59 **Total (R\$):** 533,01

[F2] - Consulta Matérias-prima

1 - Embalagens 2 - Tanques

Embalagem: _____ **Qtde.:** _____ **Total:** _____

Seq.	Código	Embalagem	Qtde.	Total (Lts)
1	2	EMBALAGEM_045	2	1,8

Qtde. Aberto (Lts) 0,5 **Qtde. Aberto (Kg)** 1

Figura 17 – Lançamento do inventário 04

5 CONCLUSÃO

Como proposto, o **Sistema de inventário de matéria-prima** conseguiu automatizar as tarefas e atender as necessidades da empresa Mecal Produtos Químicos Ltda., facilitando o trabalho da área de produção e aperfeiçoando os processos de produção da mesma.

Fiquei muito satisfeito com os resultados obtidos pois pude ampliar meus conhecimentos no que diz respeito a desenvolvimento de softwares. Além da possibilidade da utilização de ferramentas *free e open source*, que estão em constantes atualizações no mundo inteiro.

A implementação do sistema vai continuar como um trabalho futuro, implementando novas funcionalidades tentando sempre atender as necessidades propostas.

REFERÊNCIAS

LAUDON, Kenneth C., LAUDON, Jane Price. **Sistemas de informações**. 4. ed. Rio de Janeiro: Ed. LTC, 1999. 389p.

LEMAY, Laura, CADENHEAD, Rogers. **Aprenda em 21 dias Java 2**. Rio de Janeiro: Campus, 1999. 661p.

JEPSON, Brian. **Programando Banco de Dados em Java**. São Paulo: Makron Books, 1997. 487p.

NEWMAN, Alexander. **Usando Java**. Rio de Janeiro: Campus, 1997. 861p.

JOHANN, Jorge Renato. **Introdução ao método científico**. Canoas: ed. ULBRA 141p.