

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA
FACULDADE DE TECNOLOGIA DE BOTUCATU CURSO SUPERIOR DE
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

LUIS THIAGO GALLERANI PINTO

**DESENVOLVIMENTO DE *SOFTWARE WEB* DE CONTROLE E PREVISÃO
DE VENDAS PARA UMA FARMÁCIA**

Orientador: Prof. Dr. Osvaldo Cesar Pinheiro de Almeida

Coorientador: Prof. Me. José Rafael Pilan

Trabalho de Conclusão de Curso apresentado à FATEC -
Faculdade de Tecnologia de Botucatu, para obtenção do
título de Tecnólogo no Curso Superior de Análise e
Desenvolvimento de Sistemas.

Botucatu-SP

Fevereiro - 2019

DESENVOLVIMENTO DE *SOFTWARE WEB* DE CONTROLE E PREVISÃO DE VENDAS PARA UMA FARMÁCIA

WEB SOFTWARE DEVELOPMENT OF CONTROL AND FORECASTING OF SALES FOR A PHARMACY

Luis Thiago Gallerani Pinto¹, José Rafael Pilan², Osvaldo Cesar Pinheiro de Almeida³

RESUMO

O controle adequado do processo de vendas de um estabelecimento é extremamente importante e necessário para a boa gestão do negócio. A melhor forma de organizar as informações das vendas é utilizando a tecnologia da informação para gerar um sistema. O objetivo deste trabalho foi desenvolver um *software web* para gerenciar os dados das vendas do segmento de farmácias. Para o desenvolvimento deste sistema foi utilizada a linguagem Java, no ambiente de desenvolvimento NetBeans, o sistema de gerenciamento de banco de dados MySQL, juntamente com o Hibernate, que é um *framework* de persistência de dados. Como o desenvolvimento foi de um sistema web, foram utilizadas as tecnologias Java para web, o JSP e Servlets, que permitem a geração de páginas HTML dinâmicas. Foi utilizado o recurso de JasperReport para a geração de relatórios em PDF. O resultado obtido foi um sistema web funcional, que permite ao usuário todo o controle do processo de cadastro e venda de produtos, com a possibilidade da emissão de relatórios gerenciais.

Palavras-Chave: JSP. Servlet. Sistemas de Controle de Vendas. Tecnologia da Informação.

ABSTRACT

The business of the sales process is an extremely important and necessary business for the business. One way to organize sales is by using information technology to generate a system. The objective of this work was to develop software to manage sales of the pharmacy segment. For the development of this system a Java language was used in the development environment of NetBeans, the MySQL database management system, along with Hibernate, which is a data persistence framework. Like the development of a web system, JSP and Servlets were used as Java technologies for web, which allowed a generation of dynamic HTML pages. The Jasperreport feature was used for a generation of PDF graphics. The result was a functional system, which allows the user to complete the process of registering and selling products, with the possibility of issuing management reports.

Keywords: JSP. Servlet. Sales Control Systems. Information Technology.

1 INTRODUÇÃO

O mercado farmacêutico no Brasil destaca-se como um dos maiores do mundo, ocupando a sexta posição entre os maiores mercados consumidores de medicamentos. Mesmo o Brasil passando por turbulências econômicas, a expectativa é que o país alcance a quarta posição nesse cenário, ainda em 2018, ficando atrás somente dos Estados Unidos, China e Japão (CONSELHO FEDERAL DE FARMÁCIA, 2018).

Diversos fatores contribuíram para que o mercado de medicamentos no Brasil atingisse esse nível, um dos principais foi o envelhecimento da população brasileira, que vem se acentuando ano a ano. O Instituto Brasileiro de Geografia e Estatística – IBGE – estima que a população brasileira acima dos 60 anos de idade deve passar de 14,9 milhões em 2013, para 58,4 milhões até 2060, variando de 7,4% para 26,7% do total da população, devido ao constante crescimento da expectativa de vida apresentada no país (CONSELHO FEDERAL DE FARMÁCIA, 2018).

Outro fator é o investimento de empresas locais, que adquiriram capacitações no segmento de genéricos. Muitas delas passaram a desenvolver seus próprios medicamentos.

Para Hasenclever (1997) os principais motivos para a realização de pesquisa e desenvolvimento na indústria é para expansão do negócio existente, impulsionar novos negócios e ampliar capacidades tecnológicas das empresas.

A política de genéricos é incentivada pelo governo, por diversos motivos, entre eles para aumentar a concorrência, desenvolvendo a indústria local, aumentar o acesso aos medicamentos por parte da população carente e gerar mais empregos (HASENCLEVER, 1997).

Grande parte das empresas atualmente têm seus sistemas de informação, mas algumas possuem sistemas manuais ou sistemas que não atendem às suas necessidades reais. Isso pode gerar muito retrabalho. Outras são semi-informatizadas, tendo algumas vezes a utilização de vários *softwares* para a realização de procedimentos que um único software poderia realizar. Isso acaba gerando perda de tempo e até mesmo a redundância de dados, como diversos cadastros de um mesmo cliente.

Venda de produtos ou prestação de serviços é o principal objetivo da maior parte das empresas, portanto, é necessário utilizar um sistema de vendas eficiente para agilizar procedimentos, melhorar o atendimento e aperfeiçoar o controle administrativo.

Uma farmácia é uma empresa que, como muitas, geram um volume muito alto de informações, como por exemplo, os cadastros de vendas, que devem ter seus dados inseridos, armazenados, processados, controlados e disponíveis para consulta rapidamente e com toda a precisão esperada.

Com a utilização de um sistema desenvolvido sob medida para a empresa, este problema pode ser resolvido, evitando, assim, a inconsistência de dados, redundância e a perda de tempo na realização de retrabalhos.

A utilização de um sistema *web* agrega diversos valores ao sistema como:

- Mobilidade, possibilidade de acesso ao sistema de qualquer lugar conectado à internet;
- Integração, informações precisas e atualizadas sobre o que está acontecendo na empresa;
- Flexibilidade, não é preciso instalar o sistema em vários computadores caso a empresa venha a crescer;
- Redução de investimento em computadores, precisará apenas de um computador de configuração média para conectar na internet e um navegador instalado.

Com isso, este trabalho teve como objetivo principal o desenvolvimento de um *software web* para empresa farmacêutica, para o gerenciamento de vendas, produtos e clientes, com geração de relatórios gerenciais.

Em particular busca-se otimizar as operações de vendas, possibilitando a utilização em diferentes terminais, levando a informação mais rápida e precisa aos usuários através da programação *web*.

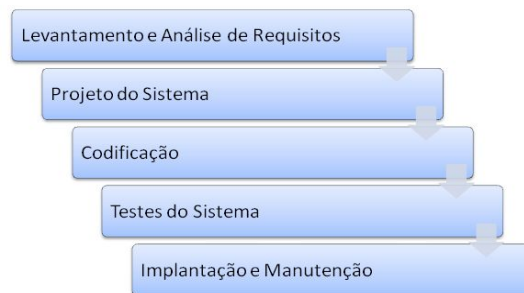
2 MATERIAL E MÉTODOS

2.1 Ciclo de Vida

O modelo clássico ou cascata, ilustrado na Figura 1, determina primeiramente o planejamento e o gerenciamento do processo de *software*. Esse modelo possui ênfase

nas fases de análise e projeto antes de partir para a programação, as fases deste modelo são: levantamento e análise dos requisitos para desenvolvimento do sistema, planejamento do projeto do sistema, codificação, realização dos testes, implantação e manutenção do sistema. Esse modelo é adotado a fim de que o objetivo do software esteja bem definido e que sejam evitados retrabalhos (PRESSMAN, 2005).

Figura 1- Ciclo de vida Clássico



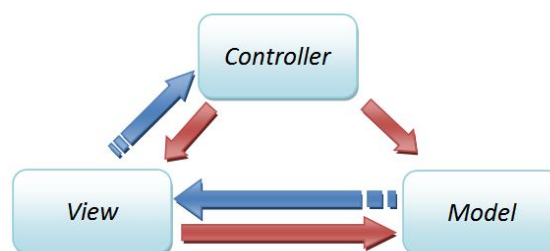
Fonte: O Autor, 2018.

Para o desenvolvimento do sistema foi empregado este modelo para priorizar as necessidades reais do sistema.

2.2 MVC

O MVC – *Model View Controller* possibilita a divisão do projeto em camadas. O *Model* é a representação do modelo de dados da aplicação. O *Controller* é a camada onde serão implementadas as regras de negócios e controle de fluxo do sistema. E a *View* representa a camada de interação com o usuário, onde serão implementadas as telas para que o usuário possa visualizar as informações (DOOLEY, 2018). O modelo é ilustrado na Figura 2.

Figura 2- MVC



Fonte: O Autor, 2018.

O principal benefício da utilização deste padrão de projeto é isolar a lógica de negócios da *interface* do usuário, desta maneira, possibilita a manutenção e reutilização de classes de forma flexível (DOOLEY, 2018). Foi utilizado este padrão de projeto para otimizar sua possível manutenção e evitar reescrita de codificações semelhantes.

2.4 WorkBench 6.3

Ferramenta para modelagem de banco de dados que integra desenvolvimento e modelagem de banco de dados, desenvolvimento e manutenção de SQL (MYSQL, 2018). Possibilitando a modelagem para dar início a implementação do banco de dados.

2.3 MySQL 8.2

MySQL é um software de banco de dados que utiliza a linguagem SQL, sendo o SQL uma linguagem de comunicação padrão com os bancos de dados. Diversas empresas utilizam este banco de dados incluindo Facebook, Google, Adobe, entre outras (MYSQL, 2018). Neste sistema foi possível cadastrar clientes, medicamentos e vendas a partir deste banco de dados.

2.5 Netbeans 8.1

Netbeans é um Ambiente de Desenvolvimento Integrado (IDE) gratuito, de código aberto e permite o desenvolvimento de softwares para desktop, móveis e web em diversas linguagens, como as linguagens Java, JavaScript, HTML5, PHP, C/C++, Ruby, entre outras (NETBEANS, 2018).

Para Gonçalves (2007), o NetBeans vem se apresentando melhor e amadurecendo a cada nova versão, e é considerado um dos melhores Ambiente de Desenvolvimento integrado (IDE) de código aberto, deixando apenas a desejar em questão de velocidade.

O uso do Netbeans facilitou a implementação dos sistemas, pois não necessita de muitas dependências para as principais funcionalidades.

2.6 Hibernate 5.2

O Hibernate é um *framework* para o mapeamento de objetos e relacionamentos de alta performance escrito na linguagem Java, ajuda na persistência de dados, evitando redundância de dados e facilita o desenvolvimento da camada Model do MVC (HIBERNATE, 2017). Com a utilização deste *framework* foi possível garantir a persistência de dados e criação do *Model*.

2.7 Servlets

Servlets são classes Java que tratam da comunicação do protocolo HTTP, sendo a base de desenvolvimento de qualquer aplicação Java web, não importando se o desenvolvimento será em *Java Server Pages (JSP)* ou *Java Server Faces (JSF)* (GONÇALVES, 2007). Todos os controles de fluxo da aplicação foram desenvolvidos utilizando esta tecnologia.

2.8 Java Server Pages

Java Server Pages (JSP) são *Servlets* embebidos em páginas HTML, ou seja, o JSP é um arquivo HTML com comandos de linguagem Java que são interpretados e convertidos em um *Servlet* em tempo de execução (GONÇALVES, 2007).

Este recurso possibilitou a criação de um *front-end* para a aplicação e foi associada com outras tecnologias abordadas, que servem não apenas para o Java, mas sim para um conjunto mais abrangente de *frameworks* e bibliotecas de desenvolvimento web.

2.9 MaterializeCSS 1.0.0

Desenvolvido pela Google, se trata de um *framework* de *design* que combina os princípios clássicos de projetos bem sucedidos junto com inovação e tecnologia, podendo ser utilizado em diversas linguagens como PHP, JSP, entre outras (MATERIALIZE, 2017). Os recursos desse *framework* foram utilizados para desenvolver um *front-end* intuitivo e limpo para o usuário final.

2.10 JQuery 3.3.1

JQuery é uma biblioteca *JavaScript* com diversas funcionalidades para a manipulação de documentos HTML, manipulação de eventos e animação que simplifica os códigos *JavaScript*, e é funcional em diversos navegadores (JQUERY, 2017). Esta biblioteca foi importante para o desenvolvimento dos recursos de interação com os usuários, como os menus e consultas.

2.11 JasperSoft 6.3

Ferramenta baseada na plataforma Eclipse que permite o desenvolvimento de relatórios em JasperReport (JASPERSOFT, 2018). O recurso de *design* da ferramenta facilita a criação de diversos relatórios, que posteriormente podem ser inseridos na aplicação. Todos os relatórios exibidos pelo sistema foram desenvolvidos utilizando esta ferramenta.

2.13 Chart.js

O Chart é uma ferramenta *JavaScript* que disponibiliza recursos para o desenvolvimento de diversos tipos de gráficos em tempo de execução (CHARTJS, 2018). O uso dessa ferramenta possibilitou o desenvolvimento de gráficos no sistema.

2.14 MD5

Se trata de uma função de algoritmo para gerar um código *hash* para criptografar uma senha. Este algoritmo foi projetado para ser rápido, simples e seguro (LUCCHESI, 1986). Foi utilizado para criptografar a senha de todos os usuários do sistema.

2.15 Regressão Linear Simples

O modelo de regressão linear segundo Krajewski, Ritzman e Malhotra (2009), é um dos modelos mais conhecidos e utilizados para previsão de demanda, que consiste de uma variável chamada de dependente estar relacionada a uma ou mais variáveis independentes por uma equação linear. Pode-se dizer em uma linguagem que a linha de regressão minimiza os desvios quadrados dos dados reais. Para obter o cálculo da equação da reta basta aplicar a seguinte equação: $y = a + bx$.

Este modelo foi utilizado para a elaboração de um algoritmo que compara os valores de vendas dos três últimos meses e exibe uma projeção de venda para o próximo mês.

2.16 Desenvolvimento de *Software*

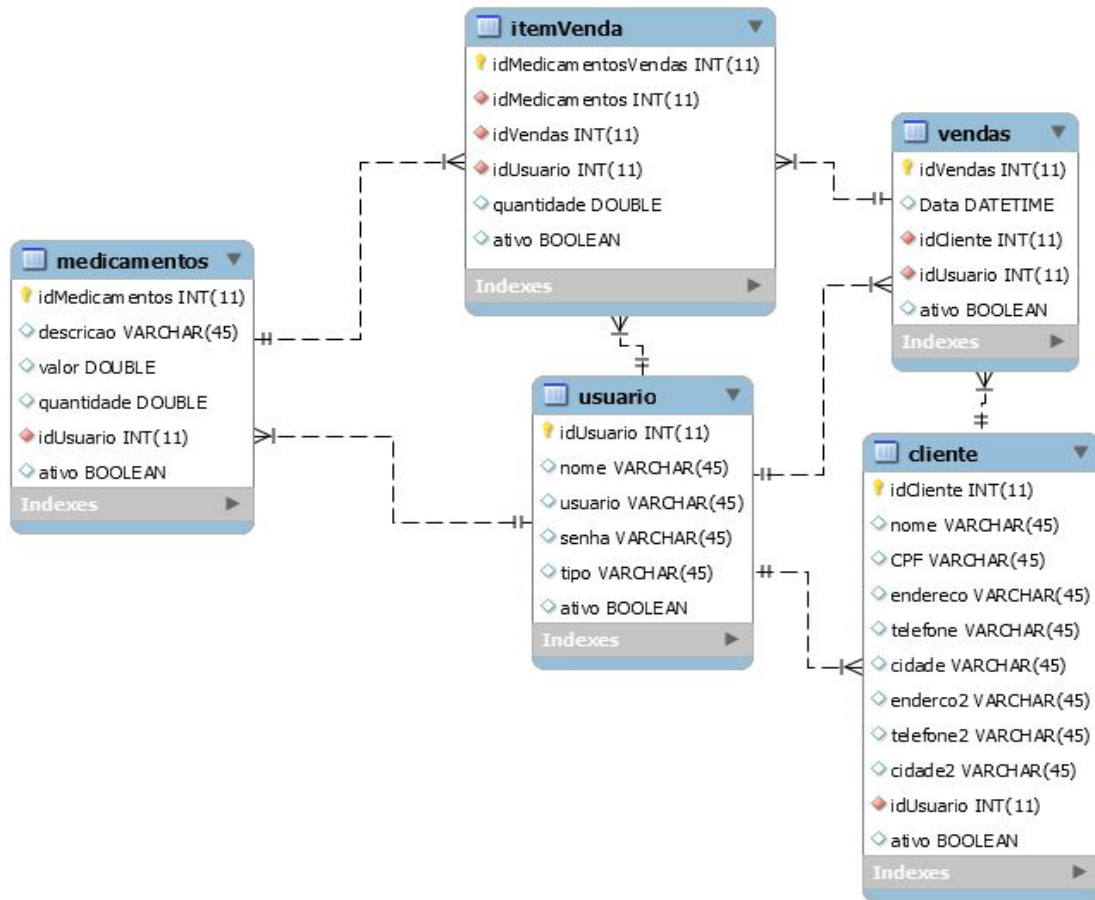
Para o desenvolvimento do projeto foi necessário realizar a coleta e análise de requisitos indo até algumas farmácias. Com os dados coletados foi elaborado um modelo de entidade e relacionamento que atendesse os requisitos necessários, modelado no WorkBench versão 6.3, para posteriormente dar origem a base de dados do sistema, onde foi desenvolvido utilizando o MySQL versão 8.0. Após a definição da base de dados iniciou o processo de implementação do sistema, usando a linguagem de programação Java versão 8 no ambiente de desenvolvimento integrado NetBeans versão 8.2. Primeiramente foi desenvolvido o *back-end* com ajuda do Hibernate versão 5.2 para mapeamento dos *Pojos*, que representam os modelos de dados, dos *Servlets*, para implementação dos controles de fluxo do sistema, e, posteriormente, o *front-end* utilizando o JSP para o desenvolvimento de interfaces, MaterializeCSS versão 1.0.0 e JQuery versão 3.3.1 (para a definição de estilos, formatação e menus de navegação). Foram usados o JasperSoft 6.3 e o Chart.js para a geração de relatórios e gráficos em tempo de execução. E por final foi implementada a regressão linear simples para realizar possíveis previsões de vendas.

3 RESULTADO E DISCUSSÃO

3.1 Diagrama Entidade-Relacionamento (DER)

O modelo de banco de dados gerado a partir das principais necessidades de uma farmácia em controlar o processo de vendas está ilustrado na Figura 3.

Figura 3- Diagrama de Entidade e Relacionamento

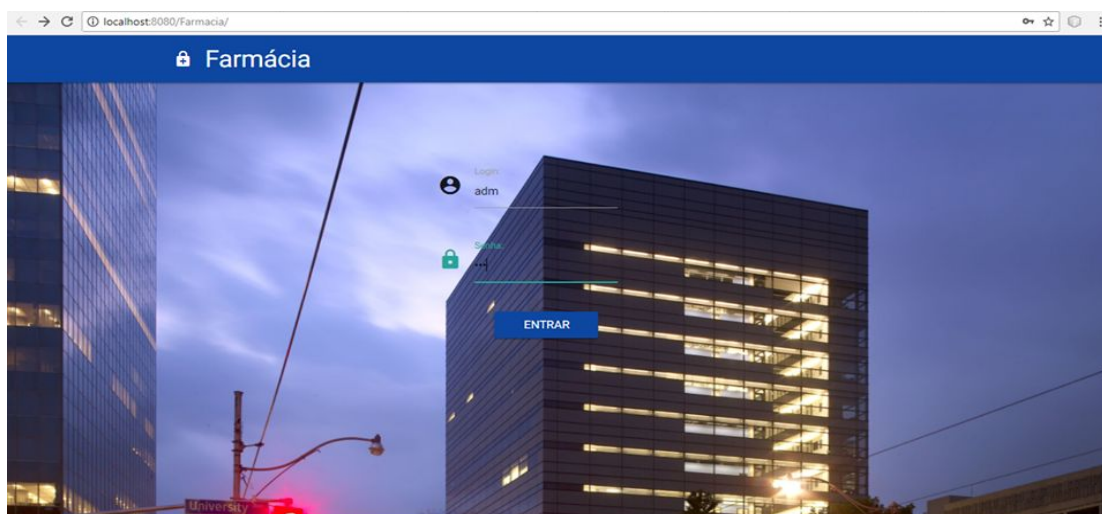


Fonte: O Autor, 2018.

3.2 Interfaces do sistema

A interface inicial do sistema (Figura 4), permite a autenticação do usuário a partir do login e senha. Caso o sistema não encontre um usuário válido ou exista uma tentativa de acesso via barra de endereços do navegador, o usuário permanecerá na página inicial do sistema. Esse recurso foi implementado utilizando um filtro, como pode ser visualizado na Figura 5.

Figura 4- Página inicial do sistema



Fonte: O Autor, 2018.

Figura 5- Codificação do filtro de segurança

```
@WebFilter(filterName = "FiltroVisao", urlPatterns = {"*.jsp"})
public class FiltroVisao implements Filter{

    @Override
    public void init(FilterConfig filterConfig) throws ServletException {

    }

    @Override
    public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
    throws IOException, ServletException {

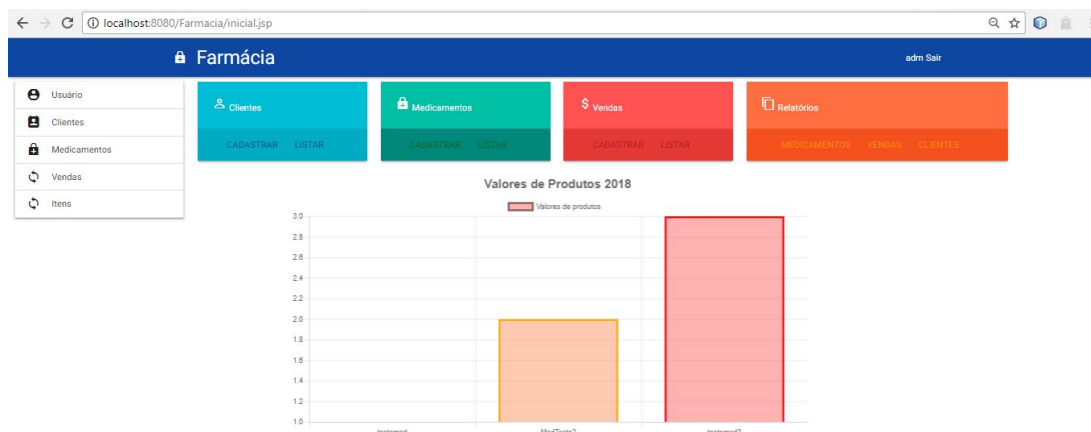
        HttpSession session = ((HttpServletRequest)request).getSession();

        Usuario usuario = (Usuario)session.getAttribute("user");
        if(usuario==null)
        {
            RequestDispatcher dispatcher = request.getRequestDispatcher("/index.jsp");
            dispatcher.forward(request, response);
        } else {
            chain.doFilter(request, response);
        }
    }
}
```

Fonte: O Autor, 2018.

Após efetuar o login, o usuário do sistema pode ter acesso às funcionalidades através do menu superior ou pelos atalhos no centro da interface principal do sistema (Figura 6).

Figura 6- Interface principal do sistema



Fonte: O Autor, 2018.

O sistema permite realizar cadastros de medicamentos (Figura 7), clientes, vendas. Caso o usuário seja administrador, há também o cadastros de usuários. Todos os cadastros seguem modelo visual similar.

Figura 7- Cadastro de medicamentos

Medicamento 1

Código: 9

Descrição: Medicamento 1

Quantidade: 20.0

Valor: 20.0

NOVO ALTERAR SALVAR

Fonte: O Autor, 2018.

No cadastro de vendas também é possível realizar o cadastro de itens por vendas, possibilitando assim a venda de diversos produtos em uma única venda. Recurso ilustrado na Figura 8.

Figura 8- Cadastro de vendas

ia/exibeVendasComIten.jsp?index=0#!

Farmácia adm Sar

Vendas

15

Código: 15

Data: 04/05/2018

Cliente: 7-Ciente 1

Situação:

Pago:

NOVO ALTERAR SALVAR

Itens Vendas

Código	Venda	Medicamento	Quantidade	Valor
15	15	Medicamento 1	1.0	R\$ 20.0

Fonte: O Autor, 2018.

Este sistema também realiza consultas das vendas, consultas de usuários e consultas de clientes (Figura 9).

Figura 9 – Consulta de Clientes

localhost:3030/Farmacia/NotaCliente.jsp

Clientes

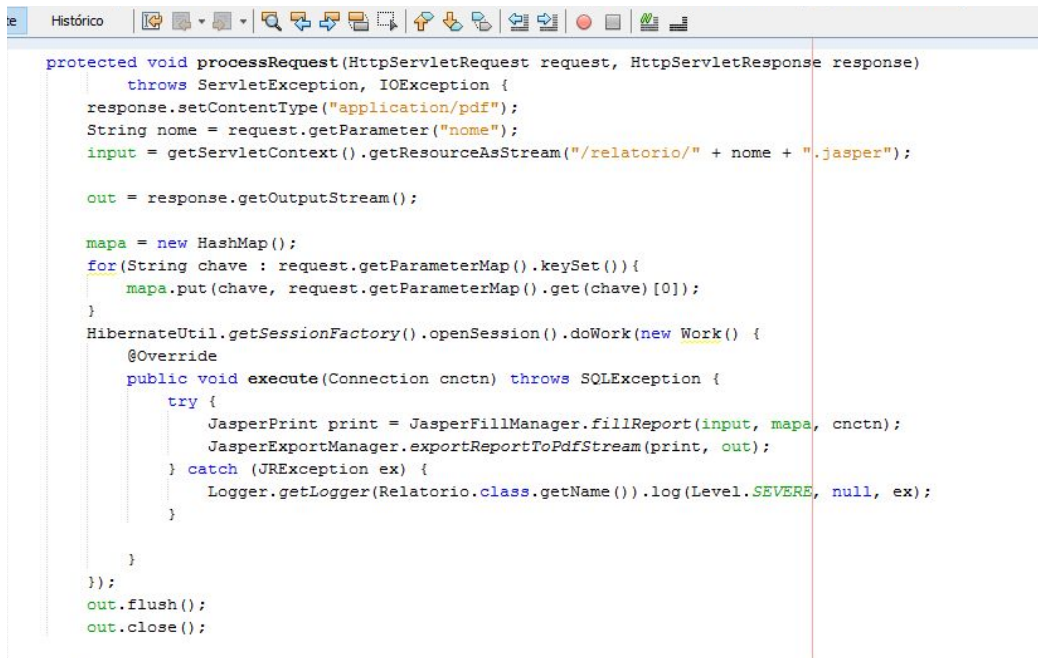
PESQUISAR

Código	Nome	CPF	Status
1	Cliente 1	36367817799	True
2	Cliente 2	36378918898	True
3	Cliente 3	36345895879	True
4	Cliente 4	36367874890	True

Fonte: O Autor, 2018.

O sistema permite gerar uma grande variedade de relatórios em PDF, como lista de clientes, usuários, vendas, gráfico de inadimplentes e recibo de pagamento, para isso é utilizado apenas um *adapter* (Servlet) (Figura 10), que se trata de um único *Controller* para várias *Views* (visualizações em PDF) que processa os relatórios.

Figura 10 - *Adapter* de relatórios



```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("application/pdf");
    String nome = request.getParameter("nome");
    input = getServletContext().getResourceAsStream("/relatorio/" + nome + ".jasper");

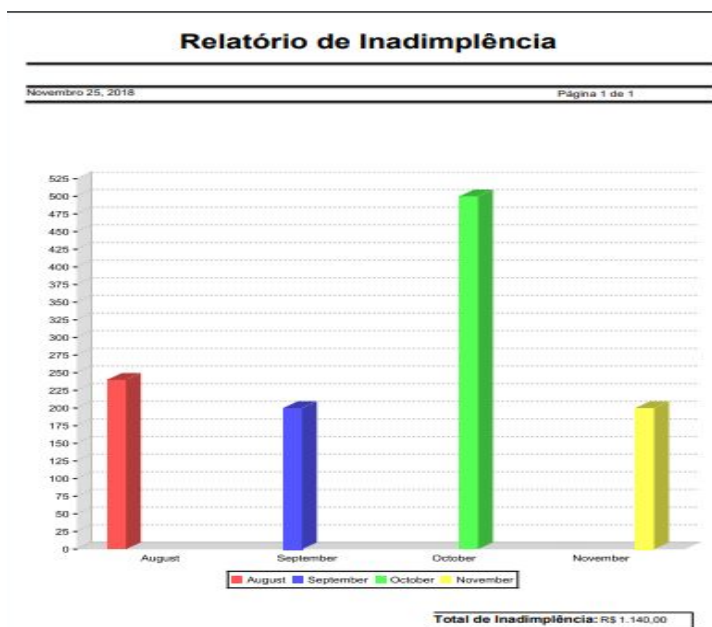
    out = response.getOutputStream();

    mapa = new HashMap();
    for(String chave : request.getParameterMap().keySet()){
        mapa.put(chave, request.getParameterMap().get(chave)[0]);
    }
    HibernateUtil.getSessionFactory().openSession().doWork(new Work() {
        @Override
        public void execute(Connection cnctn) throws SQLException {
            try {
                JasperPrint print = JasperFillManager.fillReport(input, mapa, cnctn);
                JasperExportManager.exportReportToPdfStream(print, out);
            } catch (JRException ex) {
                Logger.getLogger(Relatorio.class.getName()).log(Level.SEVERE, null, ex);
            }
        }
    });
    out.flush();
    out.close();
}
```

Fonte: O Autor, 2018.

Todos os relatórios do sistema foram desenvolvidos utilizando o JasperSoft com JasperReport, para possibilitar um design simples e facilitar a compreensão do usuário dos dados gerados. A Figura 11 ilustra um relatório de inadimplência.

Figura 11 – Relatório de inadimplência



Fonte: O Autor, 2018.

A previsão de vendas é exibida através de um gráfico para facilitar a compreensão do usuário, as duas primeiras colunas representam os meses anteriores, a terceira coluna representa o mês atual e a última coluna na cor cinza exibe a previsão de venda para o próximo mês. A previsão foi gerada aplicando a regressão linear simples. O resultado da previsão é ilustrado pela Figura 12.

Figura 12 – Previsão de Vendas



Fonte: O Autor, 2018.

6 CONCLUSÃO

O sistema mostrou-se funcional para o controle das vendas, como proposto. Por tanto, o uso desse sistema permite o gerenciamento adequado dos procedimentos de vendas de produtos farmacêuticos, facilitando a execução do processo de venda.

O uso de tecnologias como JSP, *Servlet* e Hibernate propiciaram um ganho de produtividade no processo de implementação do sistema, pois esses recursos facilitam a integração e controle dos dados com os processos de negócio implementados. Além disso, a aplicação das tecnologias JQuery, Charts e MaterializeCSS, para a criação do *front-end*, permitiu o desenvolvimento de páginas com recursos interativos e design mais intuitivo ao usuário. Por último, foi possível implementar uma previsão de vendas com dados fictícios, utilizando o modelo de regressão linear simples, no qual realiza uma análise dos últimos 3 meses e exibe uma previsão de venda para o mês seguinte, agregando mais informação e valor ao sistema .

REFERÊNCIAS

- CHARTJS. **Charts.js**. Disponível em: < <http://www.chartjs.org/docs/> >. Acesso em: 28 fev. 2018.
- CONSELHO FEDERAL DE FARMÁCIA. **Farmácias de Elite**. 2018. Disponível em: <<http://www.cff.org.br/noticia.php?id=3879>> Acesso em: 28 fev. 2018.
- DOOLEY, J.. **Software Development and Professional Practice**. 2018. Disponível em: <[https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649643\(v=pandp.10\)](https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649643(v=pandp.10))>. Acesso em: fev. 2018.
- GONÇALVES, E. **Desenvolvendo Aplicações Web com JSP Servlets, JavaServer Faces, Hibernate, EJB 3 Persistence e Ajax**. Rio de Janeiro: Editora Ciência Moderna Ltda, 2007. 736 p.
- HASENCLEVER, L. **Dinâmica e gestão da inovação: o papel das empresas industriais químicas**. Rio de Janeiro: COPPE/UFRJ, 1997. (Tese de doutorado)
- HIBERNATE. **Domain model persistence for relational databases**. 2017. Disponível em: <<http://hibernate.org/>>. Acesso em: 28 out. 2017.
- JASPERSOFT. **Jaspersoft Quick Start Guide**. 2017. Disponível em: <<https://www.jaspersoft.com/quick-start/>>. Acesso em: 28 out. 2017.
- JQUERY. **What is jQuery?** Disponível em: < <https://jquery.com/> >. Acesso em: 28 out. 2017.
- KRAJEWSKI, L. J.; RITZMAN, L. P.; MALHOTRA, M.. **Administração da produção e operações**. 8. ed. São Paulo: Pearson Prentice Hall, 2009.
- LUCCHESI, C. L: **Introdução à Criptografia Computacional**, Editora Papirus/UNICAMP, 1986.
- MATERIALIZE. **Documentation**. 2017. Disponível em: <http://materializecss.com/>>. Acesso em: 28 out. 2017.

MYSQL: MySQL Workbench. 2018. Disponível em: <<https://www.mysql.com/products/workbench/>>. Acesso em: 28 out. 2018.

NETBEANS. NetBeans IDE Features. 2018. Disponível em: <<https://netbeans.org/features/index.html>>. Acesso em: fev. 2018.

ORACLE. Java Technologies – Revolutionize Application Development. 2018a. Disponível em: <<https://www.oracle.com/java/technologies/>>. Acesso em: fev. 2018.

ORACLE. MySQL: O Banco de Dados de Código Aberto Mais Conhecido no Mundo. 2018b. Disponível em: <<https://www.oracle.com/br/mysql/index.html>>. Acesso em: 28 out. 2018.

PRESSMAN, R. S. Engenharia de software. 6. ed. São Paulo: Makron Books, 2005. 530p.