

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
CENTRO UNIVERSITÁRIO DO RIO GRANDE DO NORTE
CURSO DE ESPECIALIZAÇÃO EM DESENVOLVIMENTO DE SISTEMAS
CORPORATIVOS**

JAILSON SILVA DE OLIVEIRA

**DESENVOLVIMENTO DE UM SISTEMA *WEB* DE FOLHA DE PAGAMENTO
COMERCIAL**

**NATAL / RN
2012**

JAILSON SILVA DE OLIVEIRA

**DESENVOLVIMENTO DE UM SISTEMA *WEB* DE FOLHA DE PAGAMENTO
COMERCIAL**

Relatório Técnico apresentado ao Curso de Especialização em Desenvolvimento de Sistemas Corporativos, do Centro Universitário do Rio Grande do Norte – UNI-RN, como requisito para obtenção do título de Especialista.

Orientador: Itamir de Moraes B. Filho

Co-Orientador: Rômulo Cantanhede Fagundes

**Natal / RN
2012**

DESENVOLVIMENTO DE UM SISTEMA WEB DE FOLHA DE PAGAMENTO COMERCIAL

Relatório Técnico apresentado ao Curso de Especialização em Desenvolvimento de Sistemas Corporativos, do Centro Universitário do Rio Grande do Norte – UNI-RN, como requisito para obtenção do título de Especialista.

Aprovado em: ____/____/____

BANCA EXAMINADORA

Itamir de Moraes B. Filho (Esp)
Orientador

Prof. Gleydson Lima (MSc)
Membro Examinador

“O sucesso nasce do querer. Sempre que o homem aplicar a determinação e a persistência para um objetivo, ele vencerá os obstáculos e, se não atingir o alvo, pelo menos fará coisas admiráveis.”

José de Alencar

RESUMO

Dados, ciência e conhecimento são componentes que geram a informação unificada e mais precisa para uma empresa. Esses componentes, se separados, dificultam e tornam mais complexa a gestão e obtenção de resultados para os processos que ela gera e que têm por finalidade atingir alvos e metas de crescimento e progresso tanto para a própria unidade corporativa, quanto para as células que a formam. A tecnologia atual e suas aplicações, que armazenam e gerenciam informações, são ferramentas úteis de trabalho que fazem parte desse processo evolutivo de uma empresa, tornando a informação de fácil acesso e conhecimento para os indivíduos que a compõem e dando-lhes poder de tomar decisões coerentes com a realidade da própria instituição, a partir das informações obtidas. Sob esta ótica, o presente trabalho tem como objetivo o desenvolvimento de uma ferramenta de folha de pagamento mensal de uma empresa comercial, de forma a integrar e disponibilizar dados de uma forma dinâmica e rápida em uma base de dados, visando facilitar a administração e a manutenção dos dados de uma empresa, sejam eles dados de vendas, dados referentes ao pessoal, entre outros tipos de informações, e no intuito de atender a crescente necessidade dos gestores, explorada e descrita neste trabalho.

Palavras-chave: Dados; Informação; Sistemas de Informação; Base de Dados; Folha de Pagamento Mensal.

ABSTRACT

Data, science and knowledge are components that generate the most accurate and unified information for a company. These components, if separated, make the management of a company and the achievement of good results for the processes it generates much more complex and difficult. These processes aim to achieve targets and goals for growth and progress for both the corporate unit and the cells that form it. Today's technology and its applications, which store and manage information, are useful tools of work that are part of the evolutionary process of a company, making information and knowledge accessible to individuals who compose the company and empowering them to make the right decisions, those that are consistent with the reality of the company. Under this view, the present study aims to develop a tool for monthly payroll of a commercial enterprise, in order to integrate and deliver data in a dynamic and fast way to a database, so that the process of administration and maintenance of the data of the company can be done in a easier way, in order to meet the growing need of the managers, which will be explored and described in this work.

Keywords: Data; Knowledge; Information Systems; Database; Monthly Payroll.

LISTA DE FIGURAS

Figura 01 – Informativo de Metas Mensal	14
Figura 02 – Organograma da Empresa Ecológica Comércio Ltda	17
Figura 03 – Folha de Pagamento Mensal Feita em Planilha do <i>Microsoft Excel</i>	20
Figura 04 – Diagrama de Classes do Sistema de Folha de Pagamento Comercial ...	21
Figura 05 – Arquitetura Simplificada do <i>Hibernate</i>	29
Figura 06 – Fluxograma do Funcionamento Geral do <i>Spring Security</i>	32
Figura 07 – Arquitetura do Projeto com Seus Pacotes e Classes	36
Figura 08 – Divisão de Camadas Aplicada no Projeto Usando <i>Spring</i>	38
Figura 09 – Tela Principal do Sistema.....	39
Figura 10 – Cadastro de Funcionários	39
Figura 11 – Listagem Geral de Funcionários.....	39
Figura 12 – Lista de Funcionários Por Empresa	40
Figura 13 – Cadastro de Metas Mensal Por Loja	40
Figura 14 - Listagem das Metas Cadastradas do Mês	41
Figura 15 – Cadastro de Vendas Diárias da Loja Com Seus Vendedores	41

LISTA DE TABELAS

Tabela 01 – Cronograma do Projeto	22
Tabela 02 – Características do <i>PostgreSQL</i>	24

LISTA DE SIGLAS

ACL	Lista de Controle de Acesso
CD	Centro de Distribuição
IIS	Serviços de Informações da Internet
HTTP	<i>Hypertext Transfer Protocol</i>
JVM	<i>Java Virtual Machine</i>
LAN	<i>Local Area NetWork</i>
RH	Recursos Humanos
WEB	<i>World Wide Web</i>
VPN	<i>Virtual Private Network</i>
TEF	Transferência Eletrônica de Fundos
ACL	Lista de Controle de Acesso
WAN	<i>World Area Network</i>
ORM	<i>Object Relational Mapping</i>

SUMÁRIO

1 – INTRODUÇÃO	11
2 – OBJETIVO	18
3 – SOLUÇÃO PROPOSTA	18
4 – CRONOGRAMA	22
5 – DESENVOLVIMENTO DO PROJETO	22
5.1 – <i>POSTGRESQL</i>	22
5.2 – <i>JAVA: JDK 6.0 with Java EE – SOFTWARE</i> DE DESENVOLVIMENTO	24
5.3 – <i>TOMCAT</i>	25
5.4 – <i>IDE ECLIPSE</i>	26
5.5 – <i>Frameworks</i>	26
5.5.1 – <i>JavaServer Faces</i>	27
5.5.2 – <i>Hibernate</i>	27
5.5.3 – <i>Spring</i>	29
5.5.4 – <i>Spring Security</i>	30
6 – ESTRUTURA DO PROJETO (ARQUITETURA DE CLASSES DO PROJETO) .	32
6.1 – Fundamentos e Princípios de Desenvolvimento de Sistemas	32
6.2 – Conceitos de Coesão e Acoplamento	33
6.3 – Princípios Básicos de Programação Orientada a Objetos	34
7 – INTERFACES GRÁFICAS	39
8 – CONCLUSÃO	42
8.1 – As Vantagens do Uso de Um Sistema de Informação	42
8.2 – Sobre as Tecnologias Usadas Para Criação Deste Projeto	43
8.3 – Andamento do Sistema – Requisitos e Escopo	43
REFERÊNCIAS BIBLIOGRÁFICAS	45

1 – INTRODUÇÃO

Qual é a importância da informação para uma empresa? A informação correta, se fornecida à pessoa adequada, da forma apropriada e no tempo certo, pode melhorar e assegurar a eficiência organizacional de uma empresa. Dados, ciência e conhecimento são componentes que, quando utilizados de forma conjunta, resultam em informação, informação essa que pode ser utilizada de forma mais eficiente e favorável por meio da aplicação direcionada do conhecimento. Conhecimento esse que é requerido pelo processo de definição de relações entre dados, ou formatação de informações.

Segundo Stair *et. al.* (2005), informação é um conjunto de fatos organizados de modo a terem valor adicional, além do valor dos fatos propriamente ditos; e dados são informações compostas por fatos básicos, como nome e quantidade de horas trabalhadas em uma semana por um funcionário ou número de peças em estoque ou pedidas. Quando esses fatos básicos são organizados ou arranjados de maneira significativa, eles se transformam em informações.

Diante desse cenário, os recursos da tecnologia e o uso de aplicações que armazenam e gerenciam informações dentro das empresas são não apenas indicados, mas indispensáveis, agindo como ferramentas de trabalho contínuo de grande importância para o mercado atual, que preza pelo conceito de integração total de processos, informações e pessoas.

É nesse conceito de integração que entram os gerenciadores de banco de dados, os quais precisam de aplicações específicas que permitam a comunicação do usuário com o banco, no intuito de possibilitar a inserção e extração de informações e a integração de todos os serviços e processos existentes dentro de uma empresa.

O presente trabalho tem como objetivo apresentar e descrever o desenvolvimento de uma ferramenta de suporte e gestão de dados de folha de pagamento comercial para a empresa Ecológica Comercio Ltda, tratada neste trabalho apenas como “Ecológica”. A princípio, serão utilizados, a título de exemplificação e demonstração, apenas os perfis administrativo e comercial de

todas as filiais da empresa objeto deste trabalho. Adiante, serão demonstrados, também, os conceitos, explicações, razões e requisitos necessários para a implantação e uso do serviço em outras empresas, levando em conta suas expectativas e necessidades.

Um dos principais processos de gerenciamento das informações necessárias para a criação da folha de pagamento comercial da empresa selecionada para demonstração, a Ecológica Comercio Ltda, é o sistema de gerenciamento de metas e acompanhamento de vendas. Esse processo, realizado por meio de um formulário, arquivo individual, cujos dados ficam armazenados em uma planilha de cálculos, é um dos principais casos de uso com o qual se pretende trabalhar no desenvolvimento da ferramenta objeto deste trabalho.

O intuito do caso de uso é armazenar e gerenciar de forma simples e prática as informações de metas e vendas na modelagem do banco de dados da aplicação em uma planilha chamada “Informativo de Metas”, onde poderão ser encontrados os valores de vendas a serem atingidos por cada filial da empresa, a cada mês.

O objetivo da ferramenta de folha de pagamento comercial é, por sua vez, registrar as informações de metas e vendas de cada filial em um gerenciador de banco de dados, cuja modelagem inicial será apresentada na seção II deste trabalho, possibilitando que o gestor, por um lado, lance a informação de uma forma individual e por loja em um formulário *online* e gere relatórios e pesquisas de dados integradas e homogêneas.

Por outro lado, ela possibilitará, também, que os usuários comuns da ferramenta – funcionários das lojas da empresa – consultem e acompanhem de forma instantânea, por meio da ferramenta *web* da folha de pagamento e levando em conta a permissão de acesso estabelecida para seus usuários (*login*), suas metas a serem atingidas, que são bases para se trabalhar dentro do mês de vendas.

É interessante ressaltar que a emissão de informações diária e mensalmente a partir do lançamento e fechamento de metas de forma isolada e independente dentro de uma empresa tem como consequência a falta de um relatório integrado desses dados e, indo um pouco mais além, a falta de um histórico de pagamentos de salários de funcionários e colaboradores, para futuro acompanhamento.

Para atender essa demanda de forma prática, tanto para a gestão administrativa geral da empresa quanto para a gestão de cada loja, a ferramenta objeto deste trabalho contará com formulários fundamentais baseados nos modelos já utilizados na empresa. A título de ilustração, na Figura 01, é apresentado o *layout* do processo de lançamento de metas da forma como é realizado atualmente, por meio de planilha de cálculo do programa *Microsoft Excel*.



ECOLOGICA

Metas

NOVEMBRO 2011

Gerentes, Caixas e Vendedores

LOJA 1

05 Vendedores

1ª Quinzena (01/11 a 15/11)

	Loja	Vendedor
Meta 1	33.573,00	6.714,60
Meta 2	35.340,00	7.068,00
Meta 3	37.200,00	7.440,00
Meta 4	40.000,00	8.000,00

2ª Quinzena (16/11 a 30/11)

	Loja	Vendedor
Meta 1	40.287,60	8.057,52
Meta 2	42.408,00	8.481,60
Meta 3	44.640,00	8.928,00
Meta 4	48.000,00	9.600,00

Mensal (01/11 a 30/11)

	Loja	Vendedor
Meta 1	73.860,60	14.772,12
Meta 2	77.748,00	15.549,60
Meta 3	81.840,00	16.368,00
Meta 4	88.000,00	17.600,00

Meta V.M	120,00
Meta P.A	1.90

Gestor Administrativo

Figura 01. Informativo de Metas Mensal. Fonte: Ecológica Comercio Ltda.

Sobre a estrutura geral administrativa da empresa Ecológica, objeto de estudo deste trabalho, pode-se dizer que ela segue o modelo comum do mercado empresarial atual, centralizando a administração de suas filiais em um ponto estratégico de atendimento a sua rede.

A empresa é composta por onze filiais espalhadas na capital do Rio Grande do Norte, Natal; uma filial na praia de Pipa, no município de Tibau do Sul/RN; duas filiais na cidade de João Pessoa/PB; duas filiais na cidade de Fortaleza/CE; e, por fim, três filiais na cidade de Recife/PE, sendo todas elas integradas na tecnologia de VPN (*Virtual Private Network*).

O ponto-sede de sua administração geral, o Escritório Administrativo da Ecológica, possui um anexo conhecido como “CD”, ou Centro de Distribuição, onde se concentra todo o banco de dados da empresa relativo ao sistema de gestão de vendas e controle de estoque individual e geral das lojas. Dentro do setor de informática do Escritório Administrativo, encontra-se a infraestrutura de Tecnologia da Informação – T.I.: o servidor de banco de dados, o servidor de rede local e o gerenciamento de rede interna e da VPN de toda a rede de filiais, formando, assim, uma *intranet* tecnicamente consolidada e em uso, utilizando o recurso de transações *online* de cartões de crédito conhecido como Transferência Eletrônica de Fundos – TEF.

No que se diz respeito à hierarquia administrativa da empresa, conforme apresentado na Figura 02, encontra-se o diretor geral, proprietário da empresa; abaixo dele, o diretor financeiro, responsável pelos setores administrativos diretos: o gerente financeiro e o gerente de Recursos Humanos; abaixo dele, na hierarquia, pode-se encontrar o Departamento de Pessoal; Contas a Pagar; Gerência de Controle de Vendas com Cartões de Crédito; Centro de Distribuição e Recebimento de Toda a Mercadoria dos Fornecedores – CD; e, por fim, a administração de Tecnologia da Informação – T.I. da empresa, por onde passam e são gerenciados todos os dados e comunicação de forma computacional da empresa.

Seguindo o organograma, são encontrados dois supervisores de lojas, que gerenciam tudo que diz respeito às filiais e seus comportamentos, desde as vendas, colaboradores e atendimento ao cliente.

Nesse modelo de gestão adotado pela empresa, é possível notar o uso de diversos tipos de comunicação entre as filiais e o Escritório Administrativo, abrangendo os mais distintos temas, assuntos e setores: departamento de pessoal, contas a pagar, contas a receber oriundas das vendas da rede, controle de processos, controle de vendas por filial, informativo de metas de cada loja, envio e recebimento de *e-mails* entre as contas de usuários do domínio “ecologicabrasil.com”, entre outros.

Dentro da gestão da rede que gera os processos de comunicação e dados de controle e pessoal, podem-se encontrar dois supervisores e um gerente de Recursos Humanos. É importante ressaltar que apesar da grande importância dos processos, eles não dispõem de uma ferramenta ou sistema com uma base de dados específica e independente do sistema de gestão de vendas e estoque que permita uma melhor organização da comunicação e dos dados.

Atualmente, a folha de pagamento comercial da Ecológica é feita de forma individual e manual em planilhas de cálculos do programa *Microsoft Excel*, facilitando a ocorrência de erros e impossibilitando a realização de consultas de dados de meses e até anos anteriores de maneira única e rápida, já que não é possível obter, por exemplo, um único relatório contendo todos os funcionários da empresa agrupados mês a mês.

Para realizar uma consulta desse tipo, é preciso abrir e visualizar cada planilha de dados individualmente. Gerar relatórios de médias salariais para efeitos de pagamento de proventos que requeiram cálculos diferentes da avaliação do mês, como por exemplo, o pagamento de décimo terceiro salário e o pagamento de férias, relativos à média de um período de vendas de um determinado funcionário, torna-se um trabalho maçante, cansativo e demorado.

A ideia deste projeto é propor a integração de todos os processos relativos à folha de pagamento mensal da empresa Ecológica, bem como automatizar relatórios de informações sobre perfis de lojas, colaboradores ou quaisquer outros detalhes, permitindo a emissão da folha de pagamento mensal da empresa como um todo ou separada por filial e facilitando, assim, todo o processo de pagamento de funcionários e acompanhamento dos dados.

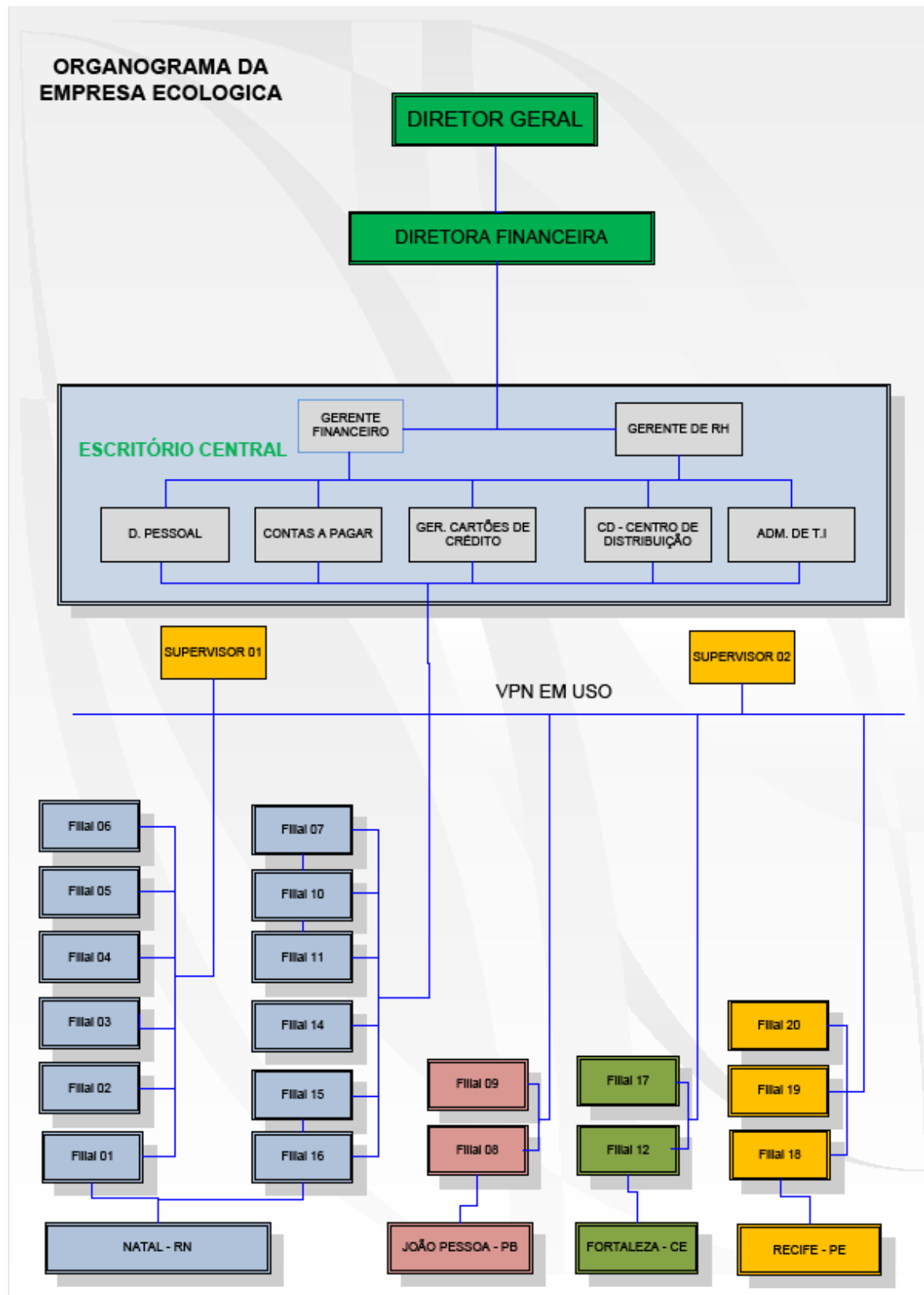


Figura 02. Organograma da Empresa Ecológica Comércio Ltda.

2 – OBJETIVO

O objetivo e interesse deste trabalho é implementar, dentro da empresa Ecológica, a aplicação modulada de folha de pagamento comercial atendendo as suas demandas, bem como integrar os dados da folha de pagamento de forma a unificar e disponibilizar informações na plataforma *web* de portabilidade de qualquer sistema operacional, recurso dentro do desenvolvimento *JAVA*, tecnologia que será apresentada detalhadamente na seção IV deste trabalho.

O projeto objetiva, também, automatizar e alinhar informações de valores, perfil e demais dados como histórico de um colaborador, seja ele por cargo ou função, no intuito de apresentar resultados instantâneos e precisos destinados não só aos comprovantes de rendimentos mensais, com também à gestão de desempenho e relatórios individuais para análise e levantamentos de rendimentos, utilizando qualquer período como parâmetro de pesquisas.

É interessante ressaltar que a ferramenta de folha de pagamento comercial foi desenvolvida a título de testes e demonstração, porém abre espaço para melhorias ao longo de sua usabilidade a partir das opiniões e *feedbacks* dos usuários administradores, que podem indicar novos recursos a serem incorporados na aplicação.

Cabe destacar que a modelagem de novas funcionalidades será de fácil acoplagem à estrutura da ferramenta já pronta e em uso, devido ao modelo de desenvolvimento aplicado, cuja metodologia será demonstrada na seção V, e ao fato da ferramenta ser uma aplicação *web*.

3 – SOLUÇÃO PROPOSTA

A aplicação de folha de pagamento comercial, objeto deste trabalho, tem a finalidade de atender o segmento varejista, no cenário da empresa Ecológica Comercio Ltda, objeto de análise e levantamento de requisitos, que tem o perfil de

uma rede de lojas, sendo cada uma dessas lojas denominadas na aplicação como "empresa". Isso porque cada filial da Ecológica tem suas próprias metas a serem alcançadas por meio de suas vendas, independentemente das demais filiais.

Cada meta a ser alcançada em uma determinada filial é rateada pela quantidade de funcionários em exercício na filial em um determinado mês, processo esse que é feito mensalmente de forma manual e individual em planilhas do programa *Microsoft Excel*. Para cada filial, tem-se um arquivo representado pelo mês de trabalho, no qual constam, no total, dezenove planilhas individuais. Dessa forma, não existe um banco de dados integrado com essas informações e valores.

A ferramenta proposta irá trabalhar em cima dessa lacuna, utilizando um banco de dados integrado onde todas as filiais poderão inserir seus dados de forma simples e prática, dados esses que estarão sempre prontamente disponíveis para acompanhamento, comparação e apresentação de relatórios de fechamento mensal, tanto por filial, quanto por agrupamento de filiais (para acompanhar o aceitação do mercado em uma determinada cidade, por exemplo) e por funcionários de acordo com suas funções.

A principal importância da aplicação que será desenvolvida, dentre outras, é a possibilidade de obter informações centralizadas em uma única base de dados, tornando possível, assim, a realização de consultas integradas. Assim, a necessidade do trabalho manual de criação de arquivos individuais com cálculos extensos e fórmulas complicadas, que facilmente induzem ao erro é totalmente eliminada.

Para ilustrar o complicado e longo processo de criação de folha de pagamento comercial em planilhas do programa *Microsoft Excel*, a Figura 03 exibe a folha de pagamento mensal atualmente utilizada na empresa Ecológica Comercio Ltda. Observando a planilha utilizada para a criação da folha de pagamento, pode-se notar que a utilização da ferramenta objeto deste trabalho viabilizará uma maior organização financeira da empresa, facilitando a gestão e o acompanhamento de seu crescimento.

	Andraza	Hugo	Amanda	Pedro	Ingrid	Venessa	Erika	Camila	Luana S.	Gildeneia	João Paulo	Isaac Severo	Raphaela	Gilmara
1º Quinzena	412,63	497,43	413,53	207,96	148,71	434,06	411,08	1.094,31					Férias	517,81
2º Quinzena	213,52	422,99	157,66	163,77	175,10	422,81	159,05							382,12
Quinquênio		34,56												91,58
Sal. Família		19,19	19,19							19,19		19,19		
Salário									540,00	540,00	540,00	540,00		1.017,60
Complemento				168,27	216,19				59,40	59,40				
Q. de Caixa									54,00	57,60				
H. Extras														
Premiação														
Dif. Retroativo														
TOTAL BRUTO	626,15	974,17	590,37	540,00	540,00	856,88	570,13	1.094,31	653,40	676,19	540,00	559,19	0,00	2.009,11
Vales	200,00	400,00	200,00	200,00	400,00		200,00	200,00	400,00	400,00		150,00		800,00
Eicha	39,49	172,40	71,99	88,40		12,30	131,70	55,30	91,00	60,90	41,50	73,99		201,45
Vale Compra														
Vale Transp	32,40	32,40	32,40	32,40	32,40	32,40	32,40		32,40	32,40	32,40	32,40		
Inss	46,07	48,84	46,07	46,07	43,20	46,07	46,07	43,20	50,68	50,68	43,20	43,20		99,82
São João														
Sindicato														
Faltas														
Egrda														
TOTAL LÍQUIDO	308,19	320,53	239,91	173,13	64,40	766,11	159,96	795,81	79,32	132,21	422,90	259,60	0,00	907,84
TOTAL BRUTO DA FOLHA		10.229,90								ABRIL/2010				
TOTAL LÍQUIDO DA FOLHA		4.629,91				Meta V.M	130,00							
TOTAL DE VENDAS		112.491,18				Meta P.A	2,00							
PORCENTAGEM DA FOLHA		9,09%												
PREMIAÇÃO	Andraza	Hugo	Amanda	Pedro	Ingrid	Venessa	Erika		Luana S.	Gildeneia	João Paulo	Isaac Severo	Nayara	TOTAL
Premiação P.A	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	60,00	60,00	60,00	60,00	0,00	240,00
Premiação V.M	0,00	120,00	0,00	0,00	0,00	0,00	0,00	0,00	60,00	60,00	60,00	60,00	0,00	360,00
Premiação V.R		300,00												300,00
FECHAMENTO MENSAL														
Meta 1	17.067,63	17.067,63	17.067,63	17.067,63	15.100,33	17.067,63	17.067,63	Meta 1	117.505,50					
Meta 2	17.965,83	17.965,83	17.965,83	17.965,83	15.895,03	17.965,83	17.965,83	Meta 2	123.690,00					
Meta 3	18.911,36	18.911,36	18.911,36	18.911,36	16.731,81	18.911,36	18.911,36	Meta 3	130.200,00					
Meta 4	20.334,81	20.334,81	20.334,81	20.334,81	17.991,11	20.334,81	20.334,81	Meta 4	140.000,00					
Vendedor(a)	Andraza	Hugo	Amanda	Pedro	Ingrid	Venessa	Erika							
Vendas	17.433,20	20.453,69	15.593,40	12.391,15	10.793,70	20.247,45	15.578,59		112.491,18					
Gerente														
TOTAL	17.433,20	20.453,69	15.593,40	12.391,15	10.793,70	20.247,45	15.578,59		112.491,18					
% Vendas	92%	108%	82%	66%	65%	107%	82%		86%					
Premiação P.A														
Premiação V.M														
FECHAMENTO QUINZENAL														
Meta 1 3.0%	9.273,80	9.273,80	9.273,80	9.273,80	7.306,60	9.273,80	9.273,80	Meta 1 0.8%	62.949,38					
Meta 2 3.5%	9.781,90	9.781,90	9.781,90	9.781,90	7.691,10	9.781,90	9.781,90	Meta 2 1.0%	66.262,50					
Meta 3 4.0%	10.275,65	10.275,65	10.275,65	10.275,65	8.096,10	10.275,65	10.275,65	Meta 3 1.2%	69.750,00					
Meta 4 4.5%	11.049,10	11.049,10	11.049,10	11.049,10	8.705,40	11.049,10	11.049,10	Meta 4 1.5%	75.000,00					
Vendedor(a)	Andraza	Hugo	Amanda	Pedro	Ingrid	Venessa	Erika		TOTAL					
Vendas	10.315,75	11.053,89	10.338,20	6.932,15	4.957,05	10.851,60	10.277,04		64.725,68					
Gerente														
TOTAL	10.315,75	11.053,89	10.338,20	6.932,15	4.957,05	10.851,60	10.277,04		64.725,68					
% Vendas	100%	108%	101%	67%	61%	106%	100%		93%					
% Comissão	4%	4,5%	4%	3%	3%	4%	4%		0,8%					
FECHAMENTO QUINZENAL														
META														
Meta 1 3.0%	7.793,73	7.793,73	7.793,73	7.793,73	7.793,73	7.793,73	7.793,73	Meta 1 0.8%	54.556,13					
Meta 2 3.5%	8.203,93	8.203,93	8.203,93	8.203,93	8.203,93	8.203,93	8.203,93	Meta 2 1.0%	57.427,50					
Meta 3 4.0%	8.635,71	8.635,71	8.635,71	8.635,71	8.635,71	8.635,71	8.635,71	Meta 3 1.2%	60.450,00					
Meta 4 4.5%	9.285,71	9.285,71	9.285,71	9.285,71	9.285,71	9.285,71	9.285,71	Meta 4 1.5%	65.000,00					
Vendedor(a)	Andraza	Hugo	Amanda	Pedro	Ingrid	Venessa	Erika		TOTAL					
Vendas	7.117,45	9.399,80	5.255,20	5.459,00	5.836,65	9.395,85	5.301,55		47.765,50					
Gerente														
TOTAL	7.117,45	9.399,80	5.255,20	5.459,00	5.836,65	9.395,85	5.301,55		47.765,50					
% Vendas	82%	109%	61%	63%	68%	109%	61%		79%					
% Comissão	3%	4,5%	3%	3%	3%	4,5%	3%		0,8%					

Figura 03. Folha de Pagamento Mensal Feita em Planilha do *Microsoft Excel*. Fonte: Ecológica Comercio Ltda.

Levando em conta o levantamento e análise de dados e requisitos, um diagrama de classes foi criado no intuito de desenvolver a arquitetura do projeto inicial visando atender suas regras de negócio. Na Figura 04, é apresentada a

representação da arquitetura inicial da aplicação, ou sistema de folha de pagamento comercial.

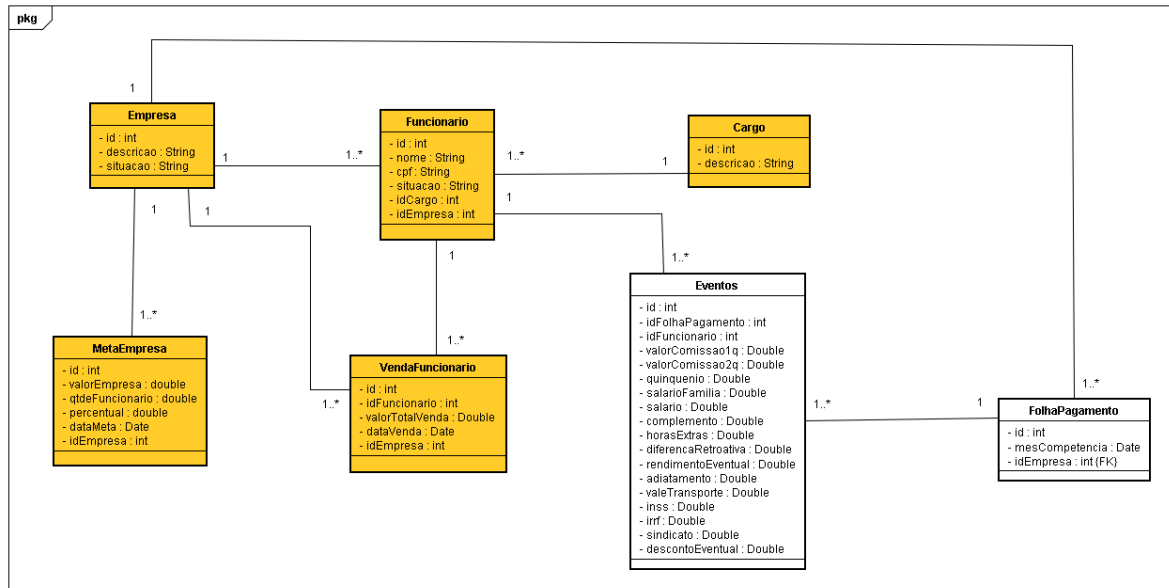


Figura 04. Diagrama de Classes do Sistema de Folha de Pagamento Comercial.

4 – CRONOGRAMA

Itens / Período	Jun	Jul	Ago	Set	Out
Idealização e preparação do projeto					
Montagem da estrutura do projeto					
Criação das principais funcionalidades					
Apresentação de slides					
					
Correções e alterações feitas pelo orientador					
Apresentação do Relatório Técnico final					

Tabela 01. Cronograma do Projeto.

5 – DESENVOLVIMENTO DO PROJETO

Este capítulo apresentará o estudo e formas de utilização de tecnologias a serem aplicadas no desenvolvimento do sistema de folha de pagamento comercial objeto deste trabalho, bem como os conceitos de aplicação e uso das mesmas.

5.1 – *POSTGRESQL*

O *PostgreSQL* é um sistema gerenciador de banco de dados objeto relacional, de código aberto, ferramenta SGBD livre que conta com amplos recursos como:

- Consultas complexas;
- Chaves estrangeiras;

- Integridade transacional;
- Controle de concorrência multi-versão;
- Suporte ao modelo híbrido objeto-relacional;
- Gatilhos;
- Visões ;
- Linguagem Procedural em várias linguagens (PL/pgSQL, PL/Python, PL/Java, PL/Perl) para procedimentos armazenagem;
- Indexação por texto;
- Estrutura para guardar dados georeferenciados *PostGIS*.

O *PostgreSQL* será utilizado para modelar, armazenar e gerenciar todos os dados coletados por meio dos formulários de entrada de dados do sistema colaborativo proposto para o caso de uso. Como um banco de dados de nível corporativo, o *PostgreSQL* possui funcionalidades sofisticadas como o controle de concorrência multiversionado conhecido como MVCC, em inglês; a recuperação em um ponto no tempo, também conhecida como PITR, em inglês; *tablespaces*; replicação assíncrona; transações agrupadas, conhecidas como *savepoints*; cópias de segurança a quente, também chamadas de *online/hot backup*; um sofisticado planejador de consultas, ou otimizador; e, por fim, um registrador de transações seqüencial (WAL) para tolerância a falhas.

O sistema gerenciador suporta conjuntos de caracteres internacionais, codificação de caracteres *multibyte*, *unicode* e sua ordenação por localização, bem como sensibilidade a caixa alta ou baixa, isto é, caracteres maiúsculos e minúsculos e formatação. É altamente escalável, tanto na grande quantidade de dados que pode gerenciar, quanto no número de usuários concorrentes que pode acomodar. Atualmente, existem sistemas ativos que utilizam o *PostgreSQL* em ambiente de produção que chegam a gerenciar mais de 4TB de dados. Alguns limites do *PostgreSQL* estão incluídos na tabela abaixo.

Tamanho Máximo do Banco de Dados	Ilimitado
Tamanho Máximo de uma Tabela	32 TB
Tamanho Máximo de uma Linha	1.6 TB
Tamanho Máximo de um Campo	1 GB
Máximo de Linhas por Tabela	Ilimitado
Máximo de Colunas por Tabela	250–1600 dependendo do tipo de coluna
Máximo de Índices por Tabela	Ilimitado

Tabela 02. Características do *PostgreSQL*. Fonte: <http://www.postgresql.org.br/sobre>.

5.2 – JAVA: *JDK 6.0 with Java EE* – SOFTWARE DE DESENVOLVIMENTO

Segundo Gonçalves (2007), *Java* é uma linguagem de programação que começou a ser desenvolvida por volta de 1991 e foi lançada no mercado para uso em 1995, pela empresa *Sun Microsystems*. A linguagem, que possui em sua estrutura desenvolvimento semelhante às linguagens C e C++, é orientada a objetos, incluindo, portanto, os benefícios de reutilização de código, extensibilidade e aplicações dinâmicas.

Um programa desenvolvimento utilizando *Java* pode ser considerado, principalmente, uma coleção de objetos em comunicação com outros objetos por meio da invocação uns dos outros. Cada objeto é de um tipo, e esse tipo é definido por uma classe ou uma interface. A maioria dos programas em *Java* utiliza um grupo de objetos dos mais variados tipos. Abaixo, segue a estrutura de programação orientada a objetos:

- **Classe** – Um modelo que descreve os tipos de estado e de comportamento que os objetos do seu tipo podem ter;
- **Objeto** – No momento da execução, quando a Máquina Virtual *Java* (*Java Virtual Machine*, ou JVM) encontra a palavra-chave “*new*”, ela usa a classe apropriada para criar um objeto, que será uma instância dessa classe. Esse

objeto terá o seu próprio estado e terá acesso a todos os comportamentos definidos pela sua classe;

- **Estado, ou Variáveis de Instâncias** – Cada objeto, ou instância da classe, terá o seu conjunto único de variáveis de instâncias conforme definido na classe. Coletivamente, os valores atribuídos às variáveis de instâncias de um objeto compõem o estado dele;
- **Métodos, ou Comportamentos** – Quando um programador cria uma classe, ele cria métodos para essa classe. É nos métodos que a lógica da classe é armazenada e o verdadeiro trabalho é realizado. Isso porque é neles que os algoritmos são executados e os dados, manipulados.

É interessante ressaltar que *Java* é uma linguagem que é primeiramente compilada e, em seguida, interpretada. Ela será utilizada como a linguagem de programação para o desenvolvimento do sistema de folha de pagamento comercial proposto.

5.3 – TOMCAT

O *Apache Tomcat* é um *container Java* e um servidor *web* ao mesmo tempo. Ele suporta a execução das tecnologias *Java Servlet* e *JavaServer Pages* (JSP), o que permite que o *Java* funcione para um ambiente *web*. É interessante destacar que, para que o *Java* funcione em aplicações escritas para *web*, é necessário ter um *Container Servlet*, que pode ser um servidor, servindo todos os tipos de aplicativos *web*, ou a integração de um, trabalhando exclusivamente para servir páginas escritas em *Java*. Atualmente, no mercado, existem diversos servidores e *containers*, sendo os mais famosos: *Apache Tomcat*, *Red Hat JBoss*, *IBM WebSphere*, etc.

Vale ressaltar que embora o *Apache Tomcat* seja robusto o suficiente para ser utilizado em ambiente de produção, ele permite integração com servidores como *Apache HTTP* e *ISS* da *Microsoft*, no intuito de fornecer uma capacidade ainda

maior de trabalho. Para a realização deste projeto, o *Tomcat* foi selecionado e aplicado como ferramenta de servidor *web*, por se mostrar umas das mais rápidas e eficientes tecnologias para esse fim atualmente disponíveis no mercado.

5.4 – IDE *ECLIPSE*

A IDE *Eclipse* é uma plataforma de desenvolvimento da linguagem *Java* considerada pela grande maioria de desenvolvedores *Java* como a melhor IDE para a linguagem. Ela conta com a característica de ser uma IDE mais rápida em relação às outras, como o *NetBeans*, por exemplo, tanto em termos de carregamento como no uso de seus assistente de código, possuindo uma grande gama de *plug-ins* e podendo, assim, se adaptar a qualquer situação de desenvolvimento.

5.5 – *FRAMEWORKS*

Para que haja uma comunicação de forma homogênea entre as linguagens de programação e as IDEs (ambiente de desenvolvimento), é necessário um elo de ligação ou uma estrutura básica. Assim acontece em desenvolvimento de *software*. Um *framework* ou arcabouço é uma abstração que une códigos comuns entre vários projetos de *softwares*, provendo uma funcionalidade genérica.

Um *framework* pode atingir uma funcionalidade específica, por configuração, durante a programação de uma aplicação. Ao contrário das bibliotecas, é o *framework* quem dita o fluxo de controle da aplicação, chamado de inversão de controle.

5.5.1 – JAVASERVER FACES

O *JavaServer*, ou JSF, é uma especificação para um *framework* de componentes para desenvolvimento *web* em *Java*. Essa especificação foi definida por meio do *Java Community Process* (JCP), que é a entidade que tem como objetivo especificar a evolução da linguagem *Java* de acordo com o interesse do mercado, e não apenas da *Sun Microsystems*, empresa criadora da linguagem. É um *framework* do mundo de *Java EE* e é desenhado para simplificar o desenvolvimento de aplicações *web*.

O uso do JSF simplifica e torna mais fácil o desenvolvimento por meio de componentes de interface de usuários (GUI), e conecta esses componentes a objetos de negócios. Também automatiza o processo de uso de *JavaBeans* e navegação de páginas.

5.5.2 – HIBERNATE

Um banco de dados relacional que utiliza uma abordagem diferente para armazenar informações, que nada se assemelha ao processo empregado em uma aplicação com base em objetos. Geralmente, o processo de persistência envolvendo objetos se dá por meio da serialização destes, de modo que essa persistência de objeto se concentraria em guardar o estado (valores) deste em disco e, em dada situação, recuperar o mesmo para fazer uso de suas informações. Essa abordagem tem uma limitação: é geralmente aplicada para persistir objetos simples (POJOs), nos quais não há relacionamentos complexos envolvendo grandes estruturas de dados.

Apesar de essa técnica poder ser aplicada a projetos maiores, seu emprego geralmente não é recomendado devido às grandes operações de escrita/leitura que podem estar envolvidas em tais processos, além da certeza da perda de desempenho. A maioria das aplicações em *Java* contém objetos que não são

persistentes. Geralmente são objetos temporários que têm um tempo de “vida” limitado pelo processo que o instanciou. Contudo, é comum em uma aplicação existir a mistura de objetos de ambas as origens, temporários e persistentes.

O *Hibernate* é *framework* ORM, isto é, *Object Relational Mapping*, ou Mapeamento Objeto-Relacional. É interessante destacar que a sigla ORM é originária de uma expressão criada antes que o uso do conceito de mapeamento de objeto/relacional se tornasse importante. De maneira direta, é possível entender ORM como uma forma automatizada e transparente de persistir objetos que pertencem a uma aplicação nas respectivas tabelas em um banco relacional, utilizando, para isso, tecnologias como *Hibernate*, que descrevem como realizar esse mapeamento entre objeto e banco de dados.

Em essência, o conceito de ORM trabalha para fazer a transformação de dados de uma forma para outra de maneira reversível. Contudo, ao tomarem-se decisões sobre a arquitetura de um sistema e ao utilizar-se essa técnica de mapeamento, é possível ter como consequência uma possível queda de desempenho no aplicativo. No entanto, o emprego do artifício traz mais benefícios que prejuízos. Basicamente, uma solução ORM contém os seguintes pontos:

- Uma API para realizar as operações CRUD básicas em objetos de classes persistentes;
- Uma linguagem ou API para especificar consultas que se referem às classes ou às propriedades das classes;
- Facilidade de especificar o método do mapeamento;
- Uma técnica para que a implementação ORM interaja com objetos transacionais, permitindo executar verificações do tipo leitura suja (*dirty checking*), ou carregamento sob demanda (*lazy association fetching*).

Apesar de ter tantas ferramentas e classes compondo seu núcleo, a arquitetura do *Hibernate* pode se ser simplificada, conforme exibido na Figura 05, que lista duas técnicas de mapeamento (*XML/annotation*). Apesar de ser possível misturar as duas, isso não é recomendado.

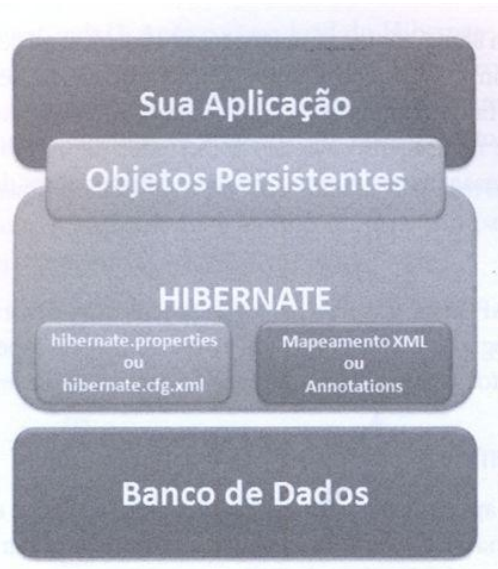


Figura 05. Arquitetura simplificada do *Hibernate*. Fonte: Livro Programação Java Para Web.

5.5.3 – SPRING

Criado por Rod Johnson, o *Spring* é um *framework* que procura dar aos seus usuários as características que possuem os *Enterprise JavaBeans* – EJB, porém sem necessitar de um *container* EJB. Seu princípio está na Inversão de Controle (*Inversion of Control*), e em sua variação denominada de Injeção de Dependência (*Dependency Injection - DI*).

Por meio do princípio na inversão de controle, é possível instanciar um conjunto de classes em uma determinada biblioteca e utilizar seus métodos. Após a chamada de um determinado método, o controle de fluxo da aplicação retorna para o usuário. O código da aplicação deve ser criado e mantido acessível ao *framework*, podendo ser através de classes que estendem outras classes pertencentes ao mesmo. O *framework*, por sua vez, realiza a chamada desse código da aplicação e, após sua utilização na aplicação, o fluxo retorna para ele.

A injeção de dependências, por sua vez, é uma forma de inversão de controle. Existem três tipos principais de injeções:

- *Constructor Injection*, ou Injeção por Construtores;
- *Setter Injection*, ou Injeção por Métodos Set;
- *Interface Injection*, ou Injeção por Interfaces.

O *Spring Framework* possui alguns módulos que trabalham cada um em uma determinada funcionalidade. São elas: *container* de DI; acesso a dados; gerenciamento de transações; agendamento de tarefas; *mail support*; *remoting*; *Spring MVC*; *Spring Web Flow*; *Spring Batch*; *Spring Web Services*; *Spring LDAP*; entre outras funcionalidades.

5.5.4 – SPRING SECURITY

O *Spring Security* é o *framework* responsável por cuidar da autenticação e autorização de usuários em aplicações *web*. A sua utilização permite que os desenvolvedores dediquem menos tempo preocupando-se com questões de segurança das aplicações, deixando-a sob total responsabilidade da ferramenta.

Ao criar uma aplicação, é importante definir inicialmente as autorizações para cada página, passo que é realizado no *Spring Security* por meio de um arquivo XML de configuração. Nesse arquivo, também devem ser descritas informações sobre o banco de dados e as consultas necessárias para obter os usuários com permissão de acesso e suas autorizações.

O *Spring Framework*, por sua vez, é um *framework Java* de código-fonte aberto, que visa facilitar o desenvolvimento *JavaEE*. Ele é baseado nos padrões de Inversão de Controle (IoC) e Injeção de Dependências. A funcionalidade básica do *Spring* é a instanciação de classes, realizando a injeção de dependências com base em definições em um arquivo de configuração XML criado pelo desenvolvedor, e resultando em um baixo acoplamento entre classes.

No entanto, o *Spring* fornece, também, outros projetos que utilizam essa funcionalidade básica, como *WebServices* com *Spring WebService*, o *Spring WebFlow*, que fornece alguns recursos para desenvolvimento de aplicativos *web* e integração com *JavaServer Faces* e *Spring Security*.

É importante frisar que a segurança é um requisito fundamental para as aplicações *web* e, portanto, as ferramentas responsáveis por garanti-la precisam acompanhar a evolução e as necessidades do mercado para assegurar que os bens mais valiosos das empresas, que são suas informações e os dados de seus clientes, continue protegido. Com isso, a *Spring Source* criou um *framework* para atender essa necessidade do mercado.

O *Spring Security* surgiu da necessidade de uma biblioteca de segurança *Java* robusta e adaptável a diversos tipos de situações. Vale ressaltar que existem diversas outras ferramentas *Java* no mercado, porém é claro afirmar que a tecnologia que a *Spring Source* oferece no *Spring Security* o torna uma opção diferenciada, com um conjunto de funcionalidades de fácil uso, além da disponibilização de apoio para integração com vários outros sistemas de autenticação.

Sempre que um usuário tentar acessar alguma pasta do sistema que tenha segurança garantida pelo *Spring Security*, será apresentada automaticamente uma tela de *login* para que o visitante possa se identificar. Caso o visitante informe o *login* e senha válida, ele será direcionado de volta à página que desejava acessar. Caso contrário, em caso de *login* e/ou senha inválidos, será exibida uma página de erro ou a própria página de *login* contendo uma mensagem informando ao visitante que seu *login* e senha não foram aceitos. Um fluxograma demonstrando o funcionamento geral do *Spring Security* é apresentado abaixo, na Figura 06.

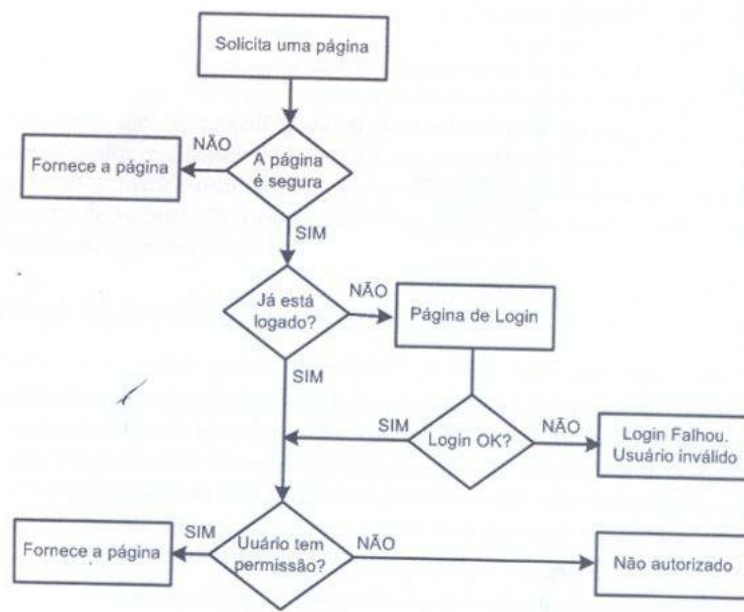


Figura 06. Fluxograma do Funcionamento Geral do *Spring Security*. Fonte: Livro Programação Java Para Web.

6 – ESTRUTURA DO PROJETO (ARQUITETURA DE CLASSES DO PROJETO)

6.1 – FUNDAMENTOS E PRINCÍPIOS DE DESENVOLVIMENTO DE SISTEMAS

Tratando-se do *design* da aplicação de folha de pagamento comercial objeto deste trabalho, foram aplicados fundamentos e princípios de desenvolvimento de sistemas, estudados e analisados no contexto referentes a padrões de desenvolvimento para aplicações orientadas a objetos.

Para obter-se uma maior qualidade no desenvolvimento de sistemas orientados a objetos, bem como uma maior facilidade de manutenção e reuso, são apresentados conceitos de Coesão e Acoplamento em sistemas orientados a objetos. O desenvolvimento de *softwares* com alta coesão e fraco acoplamento facilita, entre outras coisas, a manutenção e o reuso.

Assim sendo, o estudo desses conceitos e seus desdobramentos mostram-se não só importantes, mas indispensáveis na melhoria da qualidade de *software*. A seguir, serão destacados e explicados os conceitos necessários ao melhor entendimento de desenvolvimento da aplicação.

Os conceitos de coesão e acoplamento surgiram em meados da década de 1960, a partir de um estudo conduzido por Larry Constantine sobre o motivo pelo qual certos tipos de módulos de sistemas eram definidos e da análise dos pontos positivos e negativos relativos a estes tipos. Esses conceitos foram bastante estudados e utilizados no contexto da análise e projeto estruturado de sistemas.

6.2 – CONCEITOS DE COESÃO E ACOPLAMENTO

Coesão é um sistema de *software* como relação existente entre as responsabilidades de uma determinada classe e de cada um de seus métodos. Uma classe altamente coesa tem responsabilidades e propósitos claros e bem definidos, enquanto uma classe com baixa coesão tem muitas responsabilidades diferentes e poucas relacionadas.

Acoplamento, por sua vez, pode ser entendido como a interdependência existente entre diferentes classes. Quanto mais uma classe conhece ou depende de outras classes, maior é o grau de acoplamento entre elas. Um baixo grau de coesão geralmente acarreta em um forte acoplamento e vice-versa. Por outro lado, um alto grau de coesão resulta em um fraco acoplamento e vice-versa. Isso pode ser entendido pelo fato de que uma classe com baixa coesão possui responsabilidade que deveriam ser de outras classes e, assim sendo, deverá ser acessadas por essas outras classes para que essas possam cumprir suas tarefas, fortalecendo assim o acoplamento.

6.3 – PRINCÍPIOS BÁSICOS DE PROGRAMAÇÃO ORIENTADA A OBJETOS

Dentre todos os assuntos e contextos pesquisados durante o desenvolvimento deste trabalho, foram estudados os cinco princípios básicos da programação orientada a objetos, princípios esses que são diretivas que podem se aplicadas enquanto se trabalha em um *software* para remover problemas no código por meio da refatoração do código fonte até que ele seja legível e flexível.

Segundo Pereira (2008),

Esses princípios lidam com diversos aspectos da programação orientada a objetos como, por exemplo, como aumentar a coesão de uma classe e diminuir o acoplamento entre elas, como utilizar herança de forma correta ou como lidar com as dependências de objetos.

Os cinco princípios básicos de programação orientada a objetos e conceitos são:

- Princípio da Responsabilidade Única, também conhecido como coesão, refere-se à afirmação de que uma classe deve fazer uma única coisa e fazê-la bem;
- Princípio Aberto/Fechado, que diz que entidades de *software* (classes, módulos, funções, etc) devem ser abertas para extensões, mas fechadas para modificações. Um exemplo deste princípio é a criação de classes como interfaces e classes abstratas;
- Princípio da Substituição de Liskov, cujo conceito é um pouco mais complexo, mas compreensível na afirmação de que “se, para cada objeto o1 do tipo S, existe um objeto o2 do tipo T tal que, para todos os programas P definidos em termos de T, o comportamento de P permanece inalterado quando o1 é substituído por o2, então S é um subtipo de T”. Na prática, ele tem relação com o recurso de herança a ser utilizado em um projeto, afirmando que, se em uma alguma parte de um sistema, um objeto de uma determinada classe é requisitado, esse objeto deve poder ser substituído por qualquer objeto de

uma subclasse sem prejuízos ao funcionamento do sistema, de forma que, se um método *m* recebe como parâmetro um objeto da classe *T*, devemos poder passar para esse método qualquer objeto de qualquer classe que estenda *T* e o método deverá funcionar normalmente;

- Princípio da Segregação de Interfaces, cujo conceito afirma que clientes de interfaces não devem ser forçados a depender de métodos que eles não vão utilizar. Basicamente, ele defende que as interfaces devem se magras e não gordas. Se tivermos uma interface com 20 ou 30 métodos, é muito provável que nem todas as classes que implementam essa interface irão necessitar utilizar todos esses métodos. Isso leva à utilização de artifícios como disparar uma *UnsupportedOperationException* em métodos dessas classes;
- Princípio da Dependência de Inversão, que consiste em aplicar uma maneira de desacoplar classes de forma que as dependências convencionais entre módulos de alto nível e módulos de baixo nível sejam invertidas, de modo a tornar os módulos de alto nível independentes de detalhes da implementação dos módulos de baixo nível.

No desenvolvimento desse projeto, utiliza-se o *Spring Framework*, onde é aplicado o núcleo desse *framework* que é a injeção de dependências como forma de modelagem do projeto, usando a abordagem de inversão de controle. Segundo Paul Hammant, “inversão de controle é um *design pattern* que tem como objetivo trabalhar na resolução, configuração e ciclo de vida de uma dependência de um componente”.

Dentro do ambiente de aprendizado do Curso de Especialização em Desenvolvimento de Sistemas Corporativos, as boas práticas de desenvolvimento foram sempre abordadas e estiveram sempre presentes. Como não podia deixar de ser, o presente projeto foi desenvolvido visando atender essas boas práticas.

Na Figura 07, exibida a seguir, é apresentada a arquitetura do desenvolvimento do sistema de folha de pagamento comercial, com seus pacotes e classes partindo desse conceito.

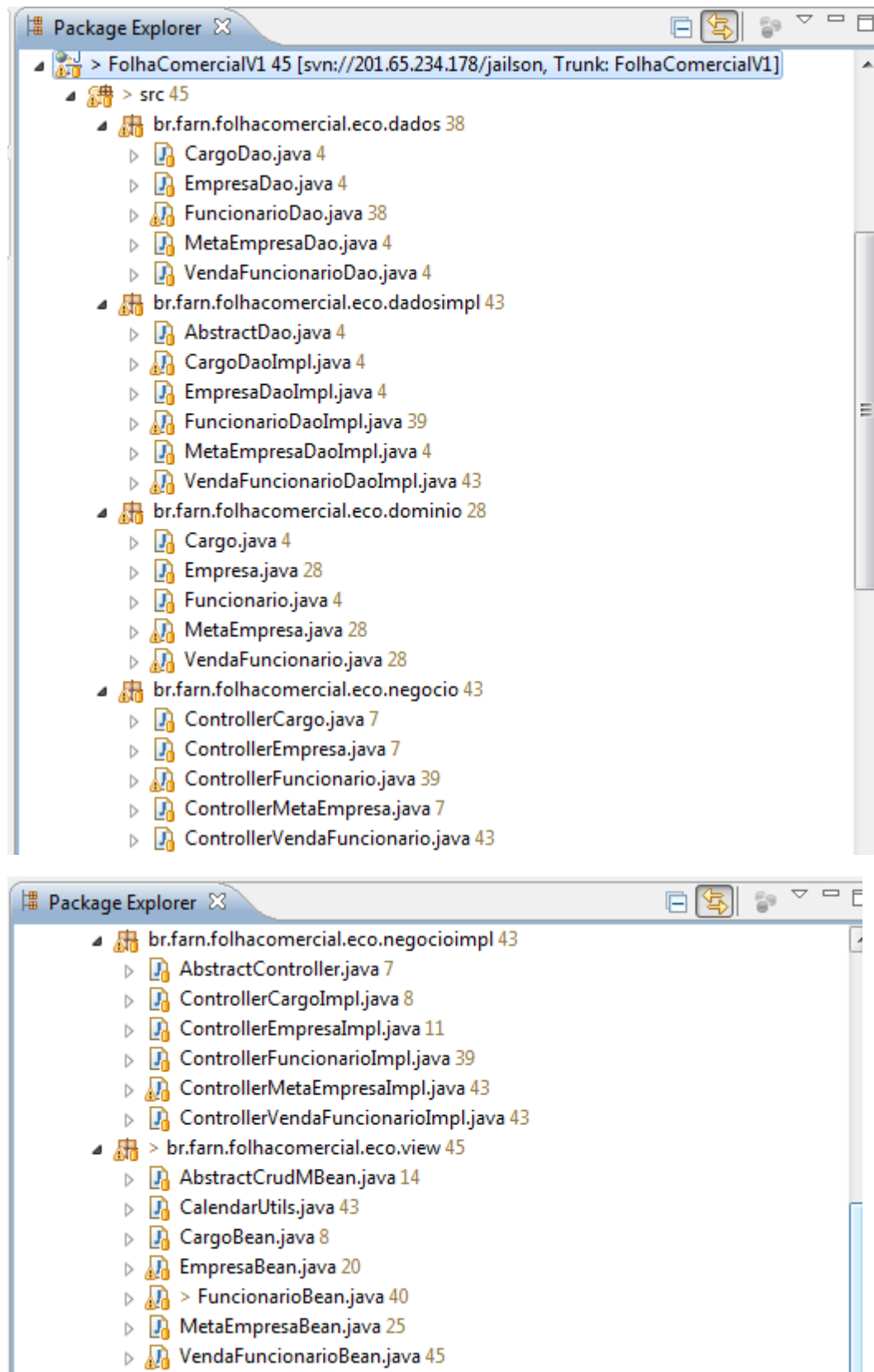
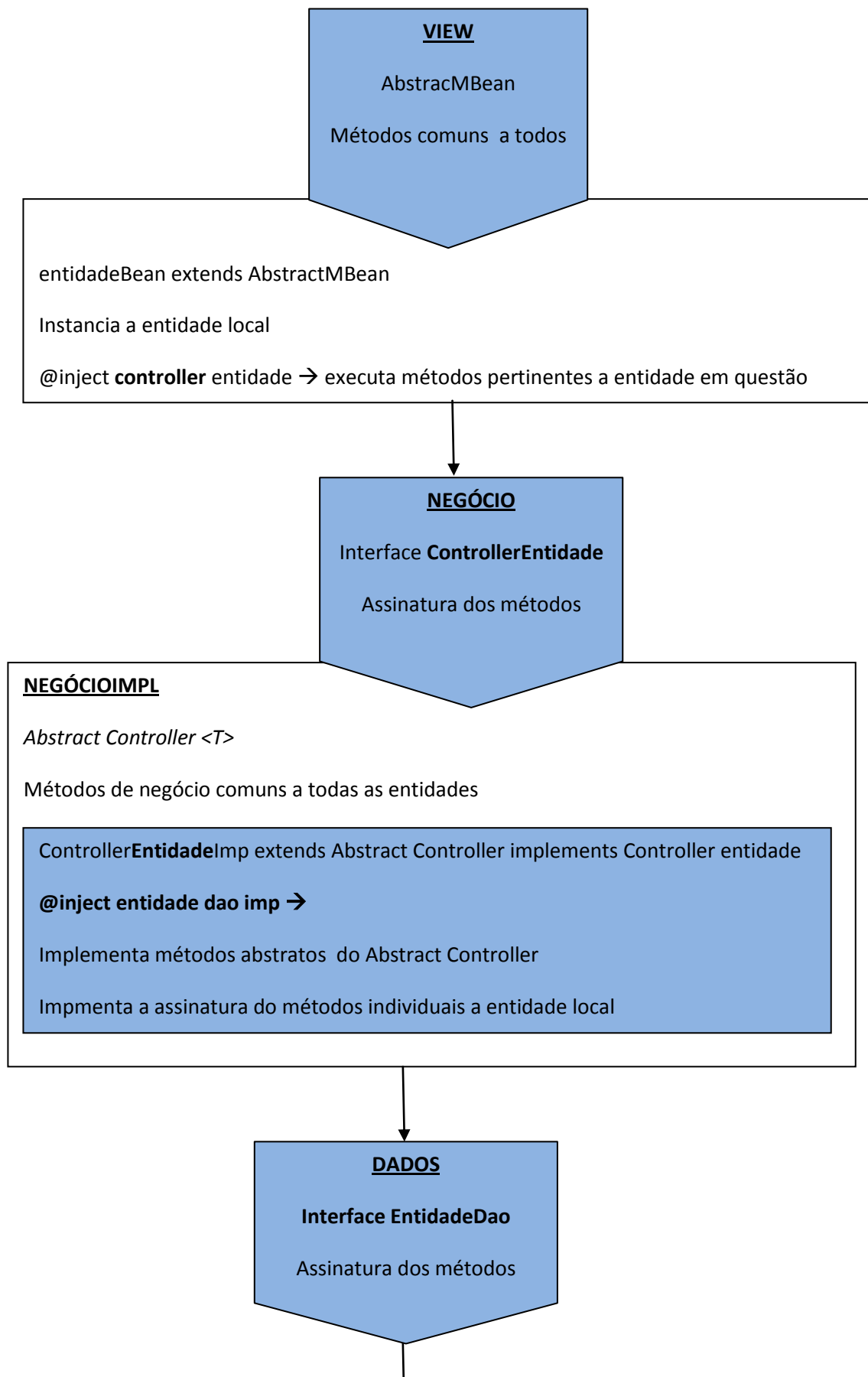


Figura 07. Arquitetura do Projeto com seus Pacotes e Classes.

Com o uso do *framework Spring*, foi definida uma divisão de camadas da aplicação seguindo o modelo MVC com *Spring*, divisão essa explanada em detalhes na Figura 06.



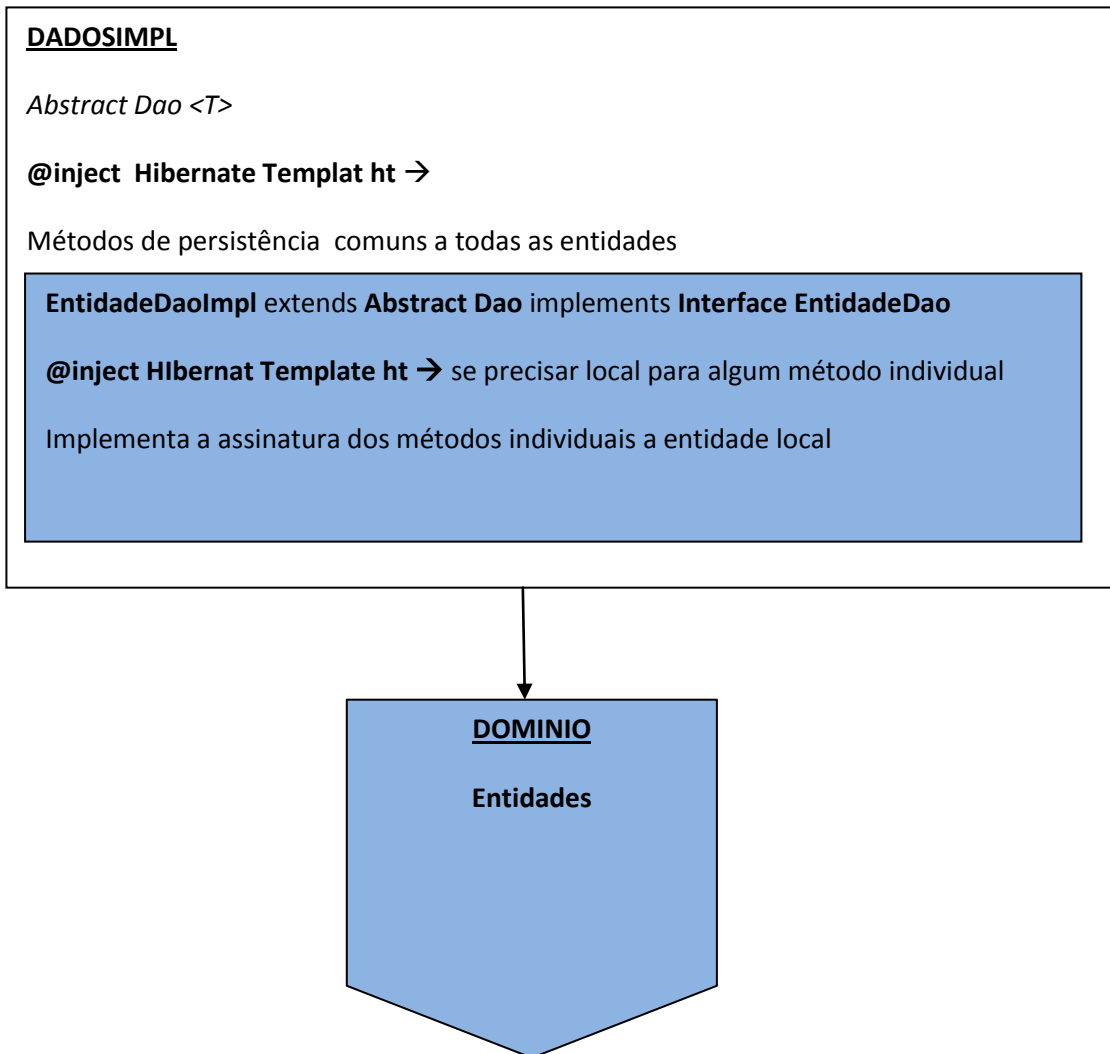


Figura 08. Divisão de Camadas Aplicada no Projeto Usando *Spring*.

7 – INTERFACES GRÁFICAS



Figura 09. Tela Principal do Sistema.

Funcionarios

Nome:

CPF:

Empresa:

Cargo:

Situação:

Figura 10. Cadastro de Funcionários.

Listagem de Funcionários

Nome	CPF	Situacao	Cargo	Empresa	Alterar	Excluir
Gabriela Glade	70217017487	A	Caixa	Empresa 001	Alterar	Excluir
Vania Sena	70217017487	A	Caixa	Empresa 002	Alterar	Excluir
Jailson Silva	33322244455	A	Caixa	Empresa 001	Alterar	Excluir
Anselmo Medeiros	44455577799	A	Caixa	Empresa 002	Alterar	Excluir
sabrina barrero de sena	33322244455	A	Caixa	Empresa 002	Alterar	Excluir

Figura 11. Listagem Geral de Funcionários.

Funcionarios

Empresa: Empresa 002 ▾

Nome	CPF
Vania Sena	70217017487
Anselmo Medeiros	44455577799
sabrina barrero de sena	33322244455

Figura 12. Lista de Funcionários Por Empresa.

Meta Empresa

Mês de referência: 01/06/2012

Empresa: Empresa 002 ▾ Qtd. Vendedores: 5.0

Meta da Loja: 480,00 Valor do Vendedor: 96,00

Valor Meta 3: 446,40 Valor Vendedor 3: 89,28

Valor Meta 2: 424,08 Valor Vendedor 2: 84,82

Valor Meta 1: 402,88 Valor Vendedor 1: 80,58

Percentual da 1 Quinzena: 40.0

1 Quinzena Loja M4: 192,00 1 Quinzena Vendedor M4: 38,40

1 Quinzena Loja M3: 178,56 1 Quinzena Vendedor M3: 35,71

1 Quinzena Loja M2: 169,63 1 Quinzena Vendedor M2: 33,93

1 Quinzena Loja M1: 161,15 1 Quinzena Vendedor M1: 32,23

2 Quinzena Loja M4: 288,00 2 Quinzena Vendedor M4: 57,60

2 Quinzena Loja M3: 267,84 2 Quinzena Vendedor M3: 53,57

2 Quinzena Loja M2: 254,45 2 Quinzena Vendedor M2: 50,89

2 Quinzena Loja M1: 241,73 2 Quinzena Vendedor M1: 48,35

Inserir MetaEmpresa **Cancelar**

Figura 13. Cadastro de Metas Mensal Por Loja.

Listagem de Metas						
Data da Meta	Empresa	Valor da Meta	Qtde. Funcionarios	Percentual da Meta	Alterar	Excluir
2012-05-31	Empresa 002	6000.0	4.0	40.0	Alterar	Excluir
2012-06-29	Empresa 001	8000.0	5.0	40.0	Alterar	Excluir
2012-06-29	Empresa 002	5000.0	4.0	35.0	Alterar	Excluir
2012-06-29	Empresa 002	8000.0	4.0	60.0	Alterar	Excluir
2012-06-29	Empresa 002	8000.0	4.0	60.0	Alterar	Excluir
2012-06-29	Empresa 001	6000.0	5.0	45.0	Alterar	Excluir
2012-06-29	Empresa 002	0.48	5.0	40.0	Alterar	Excluir
2012-05-31	Empresa 002	0.48	5.0	40.0	Alterar	Excluir

Figura 14. Listagem das Metas Cadastradas do Mês.

Vendas

Empresa: Empresa 002 ▾

Funcionários

ID	Nome	Empresa	Valor
0	Vania Sena	Empresa 002	0.0
0	Anselmo Medeiros	Empresa 002	0.0
0	sabrina barrero de sena	Empresa 002	0.0

Inserir Venda Funcionario
Listar Vendas
Cancelar

Figura 15. Cadastro de Vendas Diárias da Loja Com Seus Vendedores.

8 – CONCLUSÃO

8.1 – AS VANTAGENS DO USO DE UM SISTEMA DE INFORMAÇÃO

A necessidade de um sistema de informação integrando dados é uma realidade dentro das empresas onde se faz uso da tecnologia da informação. Elas estão se tornando cada vez mais interligadas em seus setores, principalmente no caso de empresas que utilizam o sistema de filiais que realiza as transações de vendas de produtos e escritório central com suporte para atender a demanda de dados e informações das filiais.

Levando em conta essa crescente necessidade, este trabalho idealizou a criação de uma ferramenta que tornou possível, entre várias outras ações, o gerenciamento de dados financeiros; a emissão de comprovantes de pagamento mensal; e a emissão de resultados em forma de relatórios instantâneos sem necessidade manual de procurar dados individuais.

Tudo isso de uma forma interconectada em um ambiente de rede, por meio da tecnologia *web* e com apoio de um gerenciador de banco de dados relacional, tornando o processo que antes era longo, cansativo, totalmente separado e sem gestão automática em um sistema unificado, voltado para as necessidades específicas de uma empresa.

Seguem, abaixo, algumas das vantagens possibilitadas com o uso do sistema de folha de pagamento mensal objeto deste trabalho:

- Consulta integrada de valores de salários de meses passados para um determinado funcionário;
- Consulta integrada das informações de uma ou mais filiais, individualmente ou agrupadas;
- Emissão de comprovante de pagamento ou recibo de salário mensal, integrando comissões e metas alcançadas;
- Consultas utilizando divisão de cargos ou datas passadas como parâmetros;

- Um sistema na plataforma *web*, podendo aproveitar-se dos recursos existentes dentro da própria empresa, como servidores e tecnologia pronta de VPN;
- Aumento de organização, eficiência e produtividade, possibilitada pelo acesso mais rápido à informações importantes e necessárias.

8.2 – SOBRE AS TECNOLOGIAS USADAS PARA CRIAÇÃO DESTE PROJETO

Para este trabalho, foram utilizadas, além das ferramentas de apoio ao desenvolvimento do sistema como os *Frameworks* JSF 2.0, *Hibernate*, *Spring* e *Spring Security*, todo o aprendizado adquirido no Curso Especialização em Desenvolvimento de Sistemas Corporativos, com foco em ambiente *Java*, aprendizado este que abrange desde ao estudo de Linguagem de Modelagem Unificada – UML até a conexão com banco de dados, desenvolvimento para ambientes *desktops*, desenvolvimento com JSPs, entre outras tecnologias até o ambiente proposto neste projeto.

Com isso posto, é possível concluir que o desenvolvimento do sistema proposto não foi um processo fácil ou simples, mas que se tornou possível graças aos novos conhecimentos adquiridos em toda a duração do Curso.

8.3 – ANDAMENTO DO SISTEMA – REQUISITOS E ESCOPO

Como já foi exposto anteriormente, o objetivo deste trabalho foi criar um modelo de sistema de folha de pagamento comercial para atender a necessidade da empresa Ecológica Comercio LTDA, com base em um ambiente de desenvolvimento *web*. A meta era criar uma base de dados única e uma aplicação *web* com acesso em toda a rede de filiais por meio de uma Lista de Controle de Acessos – ACL.

Toda a modelagem básica para a criação do banco de dados foi desenvolvida e, em seguida, incorporada às telas de *login* do sistema, por onde foi disponibilizado o acesso aos principais formulários de entrada de dados a serem alimentados dentro da base de dados. Todos os requisitos estão documentados nos principais casos de uso levantados na pesquisa na fase de idealização do projeto, casos de usos esses que se baseiam nos formulários que são utilizados diariamente na realização dos processos financeiros da própria empresa.

Ainda para efeito de melhorias e funcionalidade para a empresa, a aplicação de folha de pagamento comercial foi desenvolvida de forma a ser melhorada ao longo do tempo, acrescentando novos módulos voltados para a área de Recursos Humanos – RH da empresa.

REFERÊNCIAS BIBLIOGRÁFICAS

BATISTA, Emerson de Oliveira. **Sistemas de Informação: O Uso Consciente da Tecnologia Para Gerenciamento**. São Paulo: Saraiva, 2004.

FAYAD; SCHMIDT. Framework. Disponível em: <<http://pt.wikipedia.org/wiki/Framework#Refer.C3.AAncias>>. Acesso em: 30 abr 2010.

GONÇALVES, Edson. **Desenvolvendo Aplicações Web com JSP Servlets, JavaServer Faces, Hibernate, EJB3 Persistence e Ajax**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2007.

GONÇALVES, Edson. **Dominando Java Server Faces e Facelets Utilizando Spring 2.5, Hibernate e JPA**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2008.

LUCKOW, Décio Heinzelmann. **Programação Java para Web**. / Décio Heinzelman Luckow e Alexandre Altair de Melo. -- São Paulo: Novatec Editora, 2010.

LUQUE, Leandro; SILVA, Rodrigo Rocha. **Coesão e Acoplamento em Sistemas Orientados a Objetos**. Revista Java Magazine, ed. 77.

PEREIRA, David. **Desenvolvendo Software Sólido**. Revista Java Magazine, 2008.

SANTOS, Aldemar de Araújo. **Informática na Empresa**. São Paulo: Atlas, 2000.

STAIR, Ralph M.; REYNOLDS, George W. **Princípios de Sistemas de Informação**. São Paulo: Pioneira, 2005.

TURBAN, Efraim; R. Kelly Rainer; Richard E. Potter. **Administração de Tecnologia da Informação: Teoria e Prática** – Tradução de Daniel Vieira – Rio de Janeiro: Elsevier, 2005.