

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS CORNÉLIO PROCÓPIO
DIRETORIA DE GRADUAÇÃO E EDUCAÇÃO PROFISSIONAL
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS

ALESSANDRO DE OLIVEIRA

**SCE7 – UM SISTEMA DE CONTROLE DE EMPRESAS
TERCEIRIZADAS**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO
2015

ALESSANDRO DE OLIVEIRA

**SCE7 – UM SISTEMA DE CONTROLE DE EMPRESAS
TERCEIRIZADAS**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Diplomação, do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Prof. Dr. André Takeshi Endo.

CORNÉLIO PROCÓPIO
2015

AGRADECIMENTOS

Agradeço primeiramente a Deus, pois sem Ele eu não teria concluído este trabalho.

A minha esposa, que me deu apoio nos momentos de dificuldade.

Ao meu orientador Prof. Dr. André Takeshi Endo, pela sabedoria com que me guiou nesta trajetória.

Aos meus colegas de sala.

A secretaria do curso, pela cooperação.

Gostaria de deixar registrado, o meu reconhecimento à minha família, pois acredito que sem o apoio deles seria muito difícil vencer mais esse desafio.

Enfim, a todos os que por algum motivo contribuíram para a realização deste trabalho de conclusão de curso.

RESUMO

OLIVEIRA, Alessandro de. **SCE7 – Um Sistema de Controle de Empresas Terceirizadas**. 2015. 61 f. Trabalho de Conclusão de Curso (Graduação) – Tecnologia em Análise e Desenvolvimento de Sistemas. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2015.

Atualmente as grandes empresas, visando o lucro, buscam formas de se tornarem mais enxutas e reduzirem seus gastos. Uma das formas mais utilizadas é a terceirização, onde empresas focam em suas atividades principais e delegam as demais atividades para empresas especializadas. No entanto, não há nessas grandes empresas um maior controle sobre as terceirizações a fim de evitar problemas futuros. Esta monografia apresenta o desenvolvimento de um sistema Web, chamado SCE7, para o controle de empresas terceirizadas. O método de desenvolvimento ágil *Scrum* foi utilizado como referência para o desenvolvimento do sistema. Também foram utilizadas a linguagem Java, o *framework JavaServer Faces* e o sistema de gerenciamento de banco de dados MySQL. Para as fases de análise e projeto utilizou-se a UML.

Palavras – chave: Terceirização. Empresas Terceirizadas. Aplicações Web.

ABSTRACT

OLIVEIRA, Alessandro de. **SCE7 - A Control System of Outsourcing Companies**. 2015. 61 f. Trabalho de Conclusão de Curso (Graduação) – Tecnologia em Análise e Desenvolvimento de Sistemas. Federal Technology University - Paraná. Cornélio Procópio, 2015.

Currently large companies, aims for maximize profit, looking for ways to become leaner and reduce their costs. One of the most used ways is outsourcing, where companies focus on their core activities and delegate other activities to specialized companies. However, there is no control, in these large companies, over the outsourcing in order to avoid future problems. This monograph presents the development of a Web system, called SCE7, to control outsourcing companies. The agile development method Scrum was used as reference for the development of the system. We also used the Java language, the JavaServer Faces framework, and the database management system MySQL. For the analysis and design phases, UML was employed.

Keywords: Outsourcing. Outsourcing Companies. Web Applications.

LISTA DE FIGURAS

Figura 1 - Mapa conceitual da terceirização.....	15
Figura 2 - Definição de informação.	17
Figura 3 - Código de criação de tabela no banco de dados através do JPA.	21
Figura 4 - Método usando o Hibernate.....	22
Figura 5 - O MVC do JSF.	23
Figura 6 - Cadastro de Funcionários.	24
Figura 7 - Interface Eclipse.....	26
Figura 8 - Interface gráfica do Ireport.	27
Figura 9 - Diagrama de Sequência na ferramenta ASTAH.....	28
Figura 10 - Processo Scrum.....	30
Figura 11 - Diagrama de casos de uso.....	31
Figura 12 - Diagrama de classes.....	35
Figura 13 - Diagrama de Atividades.	36
Figura 14 - Diagrama de Pacotes - Arquitetura do Software.	39
Figura 15 - Diagrama de caso de uso - cadastrar empresa.	41
Figura 16 - Diagrama de classes do pacote model – cadastrar empresa.	41
Figura 17 - Diagrama de classes do pacote repositorio – cadastrar empresa.....	42
Figura 18 - Diagrama de classes do pacote infra – cadastrar empresa.	43
Figura 19 - Diagrama de classes do pacote servico – cadastrar empresa.....	44
Figura 20 - Diagrama de classes do pacote view – cadastrar empresa.	45
Figura 21 - Diagrama de classes do pacote converter – cadastrar empresa.	46
Figura 22 - Diagrama de sequência – cadastrar empresa.	47
Figura 23 - Página de Login.	49
Figura 24 - Página Home, fiscal acessando o menu cadastro.	49
Figura 25 - Página Home, fiscal acessando o menu consulta.....	50
Figura 26 - Página Home, analista acessando o menu cadastro.	50
Figura 27 - Página Home, analista acessando o menu consulta.....	51
Figura 28 - Página Cadastro de Empresas Terceirizadas.	52
Figura 29 - Página Consulta de Documento.....	53
Figura 30 - Mapa Conceitual, como SCE7 gera as pendências de documento.	55
Figura 31 - Página Home, analista acessando o menu relatórios.	55
Figura 32 - Relatório de funcionários por empresa.	56

LISTA DE QUADROS

Quadro 1 - Exemplos de Anotações para persistência no banco de dados.	20
Quadro 2 - Métricas do SCE7.	37
Quadro 3 - Sprints realizadas no desenvolvimento do SCE7.	38

LISTA DE SIGLAS E ABREVIATURAS

API	<i>Application Programming Interface</i>
ASP	<i>Active Server Pages</i>
CNPJ	Cadastro Nacional da Pessoa Jurídica
Crud	<i>Create, Read, Update and Delete</i>
FGTS	Fundo de Garantia do Tempo de Serviço
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
IBM	<i>International Business Machines</i>
IDE	<i>Integrated Development Environment</i>
INSS	Instituto Nacional do Seguro Social
JAAS	<i>Java Authentication and Authorization Service</i>
JPA	<i>Java Persistence API</i>
JSF	<i>JavaServer Faces</i>
MVC	<i>Model-view-controller</i>
OMG	<i>Object Management Group</i>
PDF	<i>Portable Document Format</i>
PHP	<i>Personal Home Page</i>
POJO	<i>Plain Old Java Objects</i>
SA	Sociedade Anônima
SCE7	Sistemas de Controle de Empresas Terceirizadas
UML	<i>Unified Modeling Language</i>
URL	<i>Uniform Resource Locator</i>
XHTML	<i>EXtensible HyperText Markup</i>
XML	<i>EXtensible Markup Language</i>
WWW	<i>World Wide Web</i>

SUMÁRIO

1. INTRODUÇÃO	10
1.1 OBJETIVOS.....	11
1.2 ORGANIZAÇÃO DO TEXTO	13
2. FUNDAMENTAÇÃO TEÓRICA.....	14
2.1 EMPRESAS TERCEIRIZADAS.....	14
2.2 APLICAÇÕES WEB.....	15
2.2.1 Sistemas de Informação	16
2.3 TECNOLOGIAS UTILIZADAS	17
2.3.1 Java.....	17
2.3.2 Servlets	19
2.3.3 JPA.....	19
2.3.4 Hibernate	21
2.3.5 JSF.....	22
2.3.6 Primefaces	24
2.3.7 UML	25
2.3.8 Eclipse	25
2.3.9 MySQL.....	26
2.3.10 JasperReports.....	27
2.3.11 Astah	28
2.4 PROCESSO DE DESENVOLVIMENTO	28
3. MÉTODO DE DESENVOLVIMENTO.....	31
3.1 ARTEFATOS.....	31
3.2 PROCESSO ADOTADO.....	37
3.3 ARQUITETURA DO SOFTWARE	38
3.4 DETALHES DE IMPLEMENTAÇÃO.....	40
4. SISTEMA SCE7	48
5. CONCLUSÃO	57
5.1 DIFICULDADES ENCONTRADAS.....	57
5.2 TRABALHOS FUTUROS	58
REFERÊNCIAS.....	59

1. INTRODUÇÃO

Atualmente as terceirizações são utilizadas em grande escala, dado que as empresas focam em sua atividade principal e contratam outras empresas para fazerem atividades secundárias. Segundo De Aquino (1998), um exemplo clássico dessa tendência são as indústrias automobilísticas, que são, em sua maioria montadoras, deixando a produção dos componentes do produto final para as empresas especializadas.

Esse relacionamento entre empresa tomadora de serviço, ou seja, que repassa serviço para outra empresa e a empresa terceirizada, que é contratada por uma tomadora de serviço, pode trazer prejuízos incalculáveis para a tomadora de serviços. Segundo a súmula 331 do Tribunal Superior do Trabalho, “o inadimplemento das obrigações trabalhistas, por parte do empregador, implica a responsabilidade subsidiária do tomador dos serviços quanto àquelas obrigações, desde que haja participado da relação processual e conste também do título executivo judicial” (TRIBUNAL SUPERIOR DO TRABALHO, 2014). Isto significa que, se uma empresa terceirizada não pagar seu funcionário de forma correta e não dar condições adequadas de trabalho, ele pode buscar a justiça do trabalho para receber o que é de direito e ainda exigir uma indenização, e se essa empresa terceirizada não tiver condições para pagar o funcionário, a empresa tomadora de serviços será responsabilizada.

Tendo em vista esse problema, empresas tomadoras, como a Açúcar e Alcool Bandeirantes SA, começaram a exigir de suas empresas terceirizadas a apresentação de vários documentos, tanto da empresa como dos seus funcionários. Aquela que está com seus documentos irregulares, pode sofrer algumas restrições como ter seu pagamento pela prestação de serviço suspenso até a regularização, ou em casos mais graves, ser multada.

O problema é que todo esse controle é feito no departamento de pessoal da empresa tomadora de serviços e é de forma manual. Os documentos são entregues ao funcionário responsável que utiliza uma planilha eletrônica para anotar a empresa e qual documento foi entregue. Porém, esses dados não ficam seguros, outras pessoas também acessam esta planilha, os dados podem ser facilmente

perdidos ou alterados de maneira errada, os arquivos podem ser corrompidos, o acesso às informações é lento e não se tem nenhum tipo de relatório.

Nesse contexto surge a criação do *software* SCE7 (Sistema de Controle de Empresas Terceirizadas). O SCE7 vem para informatizar esse processo, mantendo um histórico de todas as empresas terceirizadas, identificando quais empresas estão com documentos irregulares e quais são estes, gerando listas das empresas e seus respectivos funcionários para que se faça uma fiscalização, produzindo vários relatórios.

Este projeto será desenvolvido especificamente para a empresa Açúcar e Alcool Bandeirantes SA, uma usina que produtora de açúcar e álcool. Todos os requisitos do sistema foram levantados de acordo com as necessidades desta empresa.

1.1 OBJETIVOS

O objetivo deste trabalho é apresentar o SCE7, desenvolvido com o intuito de controlar os documentos, funcionários, equipamentos e irregularidades relacionados às empresas terceirizadas. Especificamente foram desenvolvidas as seguintes funcionalidades:

- **Efetuar *Login / Logout* no Sistema:** o acesso dos usuários ao sistema é feito por meio de *login*. O sistema possui três tipos de usuários: Fiscal, Analista e Administrador, estes possuem diferentes níveis de acesso.
- **Crud Empresa Terceirizada:** o sistema faz o gerenciamento das empresas terceirizadas.
- **Crud Proprietário:** o sistema faz o gerenciamento dos proprietários das empresas terceirizadas.
- **Crud Funcionário:** o sistema faz o gerenciamento dos funcionários das empresas terceirizadas.

- **Crud Equipamento:** o sistema faz o gerenciamento dos equipamentos das empresas terceirizadas, tais como caminhões, tratores, guinchos e colhedoras de cana.
- **Crud Irregularidade:** o sistema faz o gerenciamento de irregularidades, tais como de funcionários que estão prestando serviço na lavoura e não usam os equipamentos de proteção, quando solucionadas, essas irregularidades serão desativadas.
- **Crud Documento:** o sistema faz o gerenciamento de todos os documentos que as empresas terceirizadas devem apresentar a empresa tomadora de serviços.
- **Crud Competência:** o sistema faz o gerenciamento das competências, que são necessárias para o cadastro de entrega de documentos requisitados mensalmente. Cada competência é referente ao mês e ano do documento entregue.
- **Crud Entrega de Documento:** o sistema faz o gerenciamento de entregas de documento.
- **Crud Usuário:** o sistema faz o gerenciamento dos usuários.
- **Crud Cargo:** o sistema faz o gerenciamento de cargos dos funcionários terceiros.
- **Crud Setor:** o sistema faz o gerenciamento dos setores onde as empresas terceirizadas prestam serviço.
- **Gerar Irregularidade:** Os documentos que possuem validade, quando os estiverem vencidos, o sistema gera a irregularidade deles.
- **Gerar Pendência de Documento:** o sistema gera automaticamente as pendências de documentos que as empresas terceirizadas precisam apresentar.
- **Visualizar Documento Pendente:** o sistema mostra todos os documentos que não foram entregues pelas empresas terceirizadas.
- **Desativar Documento Pendente:** através das entregas de documentos, o sistema automaticamente desativa os documentos pendentes.
- **Cadastrar Irregularidade:** cadastro de irregularidade efetuado pelos Fiscais.
- **Consultar Proprietário:** tela de consulta de proprietários, específica para os Fiscais.
- **Consultar Funcionário:** tela de consulta de funcionários, específica para os Fiscais.

- **Consultar Equipamento:** tela de consulta de equipamentos, específica para alguns usuários que terão restrições de acesso ao sistema.
- **Gerar Relatório:** o sistema gera alguns relatórios que são acessados pelos usuários de acordo com suas permissões de acesso.

1.2 ORGANIZAÇÃO DO TEXTO

Esta monografia é organizada da seguinte forma: no Capítulo 1, foi descrito-se a introdução deste trabalho e apresentou-se às funcionalidades do sistema com breves comentários. No Capítulo 2, é apresentada a fundamentação teórica, trazendo os principais conceitos do trabalho e também as tecnologias utilizadas bem como o modelo de processo de desenvolvimento adotado. No Capítulo 3, é descrito o método de desenvolvimento, contendo os artefatos gerados, detalhes de implementação e métricas do SCE7. Também é apresentada neste capítulo a arquitetura do sistema. No Capítulo 4, são apresentadas as características do SCE7 bem como suas principais páginas devidamente explicadas, mostrando a visão que o cliente tem ao interagir com o sistema. No Capítulo 5, são apresentadas as considerações finais, bem como as dificuldades encontradas e trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Nesse capítulo são descritos os principais conceitos do trabalho desenvolvido, bem como a revisão de literatura vinculada às palavras chave desta monografia: Terceirização, Empresas Terceirizadas e Aplicações Web. Também são apresentadas as tecnologias utilizadas, bem como o processo de desenvolvimento adotado.

2.1 EMPRESAS TERCEIRIZADAS

Para entender sobre empresas terceirizadas é necessário entender o que é terceirização. Marcelino (2012, p.3) traz uma breve explicação sobre a terceirização, ele afirma que “terceirização é todo processo de contratação de trabalhadores por empresa interposta. Ou seja, é a relação onde o trabalho é realizado para uma empresa, mas contratado de maneira imediata por outra”. Nos últimos 20 anos, a terceirização se tornou um recurso essencial para a gestão de empresas, adotada na redução dos custos e na melhoria da qualidade dos produtos e serviços. Polônio acrescenta:

A terceirização nada mais é do que um processo de gestão empresarial, onde a contratante transfere para terceiros (pessoas físicas ou jurídicas) serviços que anteriormente eram executados, dentro da própria empresa, por seus empregados (POLONIO, 2000, p. 97).

Na Figura 1, o mapa conceitual evidencia que uma empresa tomadora de serviços, que possui um serviço a ser terceirizado para uma empresa interposta, possui um contrato de prestação de serviços. A empresa interposta, para realizar esse serviço na tomadora de serviços terá de contratar funcionários. Assim é necessário realizar um contrato de trabalho com esses funcionários. Com isso temos funcionários trabalhando como terceiros, ou seja, eles trabalham em uma empresa que tem o serviço, mas não são funcionários dessa empresa e sim de uma empresa interposta.

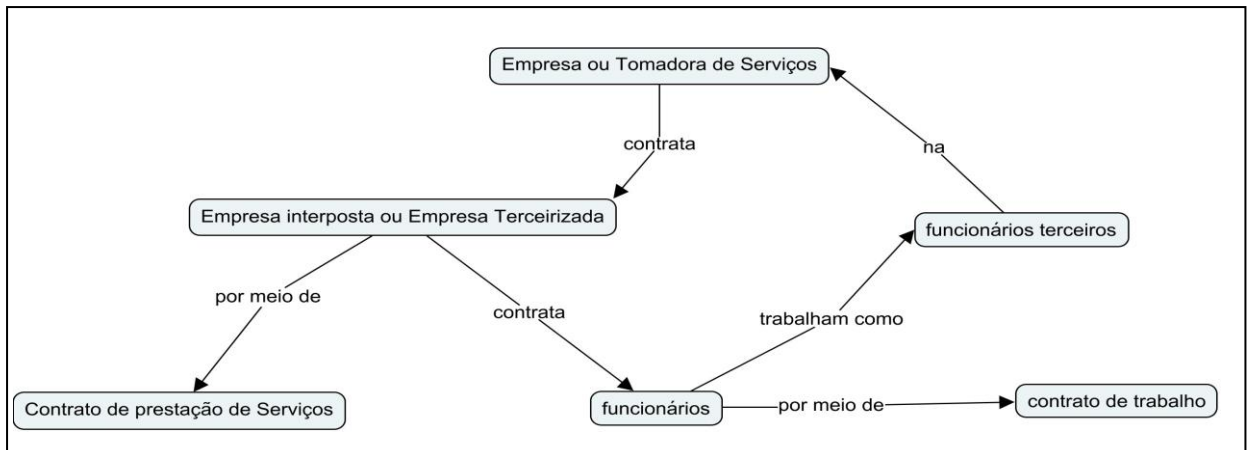


Figura 1 - Mapa conceitual da terceirização.
Fonte: Autoria Própria.

No contexto da terceirização no Brasil, o conceito de responsabilidade subsidiária é importante. Segundo o Tribunal Superior do Trabalho (2015), no inciso IV da Súmula 331, quando a empresa terceirizada não paga as obrigações trabalhistas de seu empregado, a empresa tomadora de serviços tem a obrigação de pagar. “A responsabilidade subsidiária ocorre somente pelo inadimplemento das obrigações trabalhistas da prestadora de serviços” (BERNARDES; FREIRE, 2012, p.3). Suponha que um funcionário da empresa X (uma empresa terceirizada) está prestando serviço na empresa Y (tomadora de serviços), a empresa X recebe os pagamentos pela prestação de serviços na empresa Y corretamente, porém, quando termina o serviço, a empresa X demite seu funcionário e não paga as verbas trabalhistas. Com isso o funcionário irá procurar seus direitos. O juiz do trabalho julga o caso e condena a empresa X a pagar cinquenta mil reais, caso a empresa X não tenha condições de pagar esse valor, o Juiz do trabalho irá intimar a empresa Y (tomadora de serviços), obrigando-a a pagar tal valor.

2.2 APLICAÇÕES WEB

Segundo Oliveira et al. (2005), aplicação Web é um software que implementa qualquer Sistema de Informação segundo o paradigma Cliente/Servidor suportado pelo protocolo de comunicação HTTP e possui a escrita em HTML em sua camada interativa de modo que a interface com o utilizador seja assegurada pelos

browsers tradicionalmente criados para navegação na rede de hiperdocumentos WWW.

De acordo com Alldreams (2015), Aplicação Web é um sistema desenvolvido para ser executado em um navegador, como exemplo Internet Explorer, Mozilla Firefox, entre outros. Também pode ser acessado pela Internet ou em redes internas (intranet). Algumas das vantagens de se desenvolver um sistema Web são:

- As tarefas executadas pelas aplicações Web podem ser distribuídas em uma rede de computadores. Essas aplicações Web podem ser desenvolvidas para diversas áreas, tais como portais, *webmails*, *e-commerce* e até mesmo aplicativos comerciais, possibilitando assim o acesso de qualquer lugar que possua internet e um navegador web.
- As tecnologias têm ajudado as empresas a se aproximarem mais de seus clientes.
- Os sistemas Web têm a capacidade de execução em diversos tipos de computadores. Atualmente existem aplicações altamente interativas, de fácil usabilidade e acessibilidade.

2.2.1 Sistemas de Informação

A Figura 2 define o conceito de informação. Uma informação é gerada por dados isolados que passam por um processo de classificação, filtragem e organização. Tem-se como resultado desse processo uma informação. De acordo com Côrtes (2008, p.6) “A partir do momento em que dados são transformados em informações, decisões podem ser tomadas”.

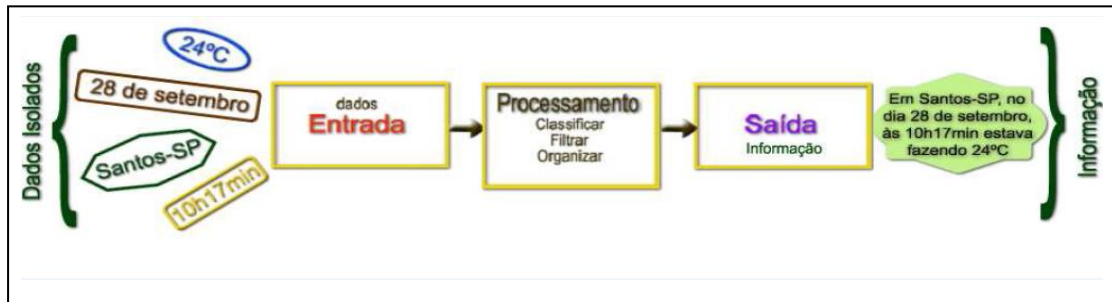


Figura 2 - Definição de informação.
Fonte: (CÔRTEZ, 2008, p.6).

Sistema é um conjunto de partes que se interagem e são interdependentes que, conjuntamente, formam um todo unitário com determinado objetivo e efetuam determinada função (OLIVEIRA, 2002). Para se ter um sistema é necessário então obter os dados, ter o conhecimento dos mesmos definindo o objetivo a ser alcançado, para a realização desse objetivo tem-se um sistema.

Diante do anteriormente exposto é obtido sistemas de informação. Um sistema de informação é um conjunto de meios e procedimentos que possui a finalidade de assegurar a informação útil necessária às várias funções e níveis das organizações, envolvendo também a parte externa delas (SISTEMASINFORMAÇÃO, 2015).

2.3 TECNOLOGIAS UTILIZADAS

Nessa seção, serão apresentadas as tecnologias utilizadas para o desenvolvimento do SCE7.

2.3.1 Java

No início da década de 90, evoluir o poder da computação em rede para as atividades da vida cotidiana era uma visão radical (ORACLE, 2015). Em 1991, um grupo de engenheiros da Sun chamado “*Green Team*” ou Equipe Verde, acreditaram

que a próxima onda na computação era a união de dispositivos de consumo digitais e computadores. Liderado por James Gosling, os engenheiros trabalharam o tempo todo e criaram uma linguagem de programação: Java (ORACLE, 2015).

Java é uma linguagem de programação que utiliza o paradigma de orientação a objetos e sua tecnologia é adotada em *laptops*, *datacenters*, consoles de jogos, supercomputadores científicos, telefones celulares e Internet (JAVA, 2015). O Java pode ser encontrado em diversos lugares (JAVA, 2015):

- 97% dos *desktops* corporativos executam o Java
- 89% dos Computadores nos EUA executam Java
- Existem 9 milhões de desenvolvedores de Java em todo o mundo
- A primeira escolha para os desenvolvedores
- Primeira plataforma de desenvolvimento Nº 1
- 3 Bilhões de telefones celulares executam o Java
- 100% dos *Blu-ray Disc Players* vêm equipados com o Java
- 5 bilhões de placas Java em uso
- 125 milhões de aparelhos de televisão executam o Java

Existem várias tecnologias na atualidade para o desenvolvimento de aplicações, onde as mais conhecidas são o PHP, ASP, ASP.NET e Python. Se comparada a essas tecnologias o Java é considerado complexo. Para justificar a ampla utilização dessa linguagem no Brasil destacam-se algumas características (GONÇALVES, 2007):

- **Portabilidade:** A aplicação desenvolvida pode ser executada em diversas plataformas, Windows, Linux, Mac OS, Android e iOS, são exemplos de plataformas aceitas.
- **Orientação a Objetos:** A programação é orientada a objetos, com esse método os sistemas complexos são desenvolvidos com mais facilidade.
- **Flexibilidade:** O Java já se encontra bastante difundido, pois possui uma enorme comunidade de desenvolvedores, diversas bibliotecas e códigos disponíveis, onde o desenvolvedor poderá usufruir sem a necessidade de gasto algum. A esse meio temos as famosas bibliotecas prontas e *frameworks*.

- **Escalabilidade:** A arquitetura de servidores desenvolvidos em Java possui alguns benefícios como escalabilidade, onde em grande parte dos servidores modernos é possível a distribuição de carga de processamento sendo que servidores podem ser adicionados e removidos acompanhando o aumento ou decréscimo dessa carga de processamento.

Esta linguagem foi utilizada no desenvolvimento do SCE7 pelos seguintes motivos: as vantagens anteriormente expostas e por ser uma linguagem já consolidada no mercado brasileiro concebendo grandes oportunidades no futuro.

2.3.2 Servlets

Servlets são classes Java, desenvolvidas a partir de uma estrutura bem definida e quando instaladas em um servidor, permitem a geração dinâmica de conteúdo como HTML, além de tratar requisições recebidas de clientes. Servlets são a base das aplicações desenvolvidas em Java para a Web (GONÇALVES, 2007). Neste trabalho, como foi utilizado o *framework JavaServer Faces*, os Servlets são executados de forma implícita.

2.3.3 JPA

JPA, sigla para *Java Persistence API*, é baseada no conceito *Plain Old Java Object* (POJO). Nada mais é do que a incorporação de renomados *frameworks* de persistência para padronizar o mapeamento objeto relacional em Java (GONÇALVES, 2007).

Na JPA, os objetos persistentes são denominados entidades (*Entities*). Uma entidade é um objeto simples no qual irá representar um conjunto de dados persistido no banco de dados. As entidades são definidas por classes Java comuns, sem o uso de *frameworks* ou bibliotecas, por isso elas podem ser abstratas ou herdar de outras classes sem nenhuma restrição (GONÇALVES, 2007).

As classes e interfaces da JPA estão acessíveis no pacote de bibliotecas do Java `javax.persistence`. Com isso, é possível fazer o mapeamento de uma aplicação utilizando anotações. Anotações são formas específicas de declarações de metadados que podem ser incluídas ao código-fonte pelo programador (SANTOS, 2005). Elas proveem informações sobre o comportamento de um programa. No Quadro 1, é exemplificado as anotações da JPA:

Anotação	Descrição
@Entity	Representa uma entidade, sendo ela uma classe Java comum.
@Table	Representa uma tabela no banco de dados.
@Id	Representa a chave primária de uma tabela.
@Column	Representa uma coluna de uma tabela.
@OneToOne	Significa que a tabela possui uma relação de 1 para 1 com outra tabela.
@OneToMany	Significa que a tabela possui uma relação de 1 para n com outra tabela.
@ManyToOne	Significa que a tabela possui uma relação de n para 1 com outra tabela.
@ManyToMany	Significa que a tabela possui uma relação de n para n com outra tabela

Quadro 1 - Exemplos de anotações para persistência no banco de dados.
Fonte: Gonçalves (2007).

Por meio da tecnologia JPA, o SCE7 foi desenvolvido sem trabalhar diretamente com o banco de dados, ou seja, por meio de anotações foi possível fazer toda a persistência no banco, como criação de tabelas com suas respectivas colunas, bem como toda a parte de relacionamento entre as tabelas diretamente na linguagem de programação. A Figura 3 apresenta um exemplo de código que cria uma tabela no banco de dados através do JPA:

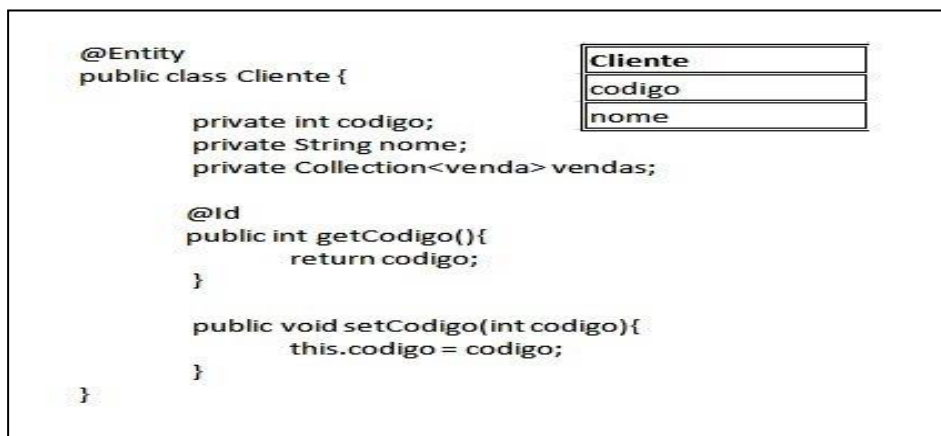


Figura 3 - Código de criação de tabela no banco de dados por meio do JPA.
Fonte: (OLIVEIRA, 2014).

Na Figura 3, é criada uma classe cliente com as variáveis código do tipo inteiro, nome do tipo *String* e uma coleção de vendas. Em relação as anotações, o `@Entity` define a classe Cliente como entidade, no banco de dados ela se tornará uma tabela. O `@Id` está indicando que o codigo será o identificador dessa tabela no banco. Com essas anotações é possível criar o banco de dados sem trabalhar diretamente com ele.

2.3.4 Hibernate

É um *framework* usado para facilitar todo o processo de persistência no banco de dados, reduzindo tempo de desenvolvimento e o custo dos *softwares*. Além de ter sua própria API nativa, o Hibernate implementa a especificação JPA, é por causa dessa característica que vários sistemas utilizam o JPA juntamente com o Hibernate (HIBERNATE, 2015). Na Figura 4, é apresentado um método do SCE7 usando o Hibernate.

```
public Cargo comDadosIguais(Cargo cargo) {  
    return (Cargo) this.session.createCriteria(Cargo.class)  
        .add(Restrictions.ilike("nome", cargo.getNome()))  
        .uniqueResult();  
}
```

Figura 4 - Método usando o Hibernate.
Fonte: Autoria Própria.

O método citado na Figura 4 é responsável por não permitir que o sistema salve cargos com nomes iguais no banco de dados. Quando o usuário preenche o formulário com dados do cargo e salva o mesmo, antes que a informação seja persistida no banco, o SCE7 executa o método `comDadosIguais` que recebe o cargo a ser persistido como parâmetro e por meio da linguagem do Hibernate executa uma consulta no banco de dados. Esta consulta irá retornar um cargo que tenha o mesmo nome do cargo a ser persistido, com isso o sistema não efetua a persistência do cargo e avisa o usuário que já existem os dados no banco, se não existir o cargo no banco de dados o sistema salva o cargo normalmente.

2.3.5 JSF

JSF, sigla para *JavaServer Faces* é um *framework* de componentes de interface de usuário (UI) para aplicações Web desenvolvidas em Java. Ele foi criado para ajudar significativamente a escrita e manutenção de aplicações que executem em um Servidor Java e apresentam suas interfaces de usuários de volta para um cliente específico (JACOBI; FALLOWS, 2007).

De acordo com Gonçalves (2007), é de responsabilidade do *framework* JSF a interação com o usuário e o fornecimento de ferramentas para criar a apresentação visual, a aplicação lógica e a lógica de negócios de uma aplicação Web. Conforme Jacobi e Fallows (2007) o *framework* JSF é baseado na arquitetura *Model, View, Controller* (MVC):

- **Model (Modelo):** Com o JSF, o conceito de *managed bean* (*bean* gerenciado) foi desenvolvido. Os *managed bean* são definidos no arquivo `faces-config.xml` e

oferecem ao programador acesso total a todos os métodos mapeados do *bean* de apoio, objetos armazenados e regras de negócio.

- **View (Visão):** A camada *view* do JSF descreve o layout desejado, o comportamento e a renderização da aplicação. Uma das peças-chave de um aplicativo em JSF é o *UIComponent* que por sua vez é a base da camada *view*.
- **Controller (Controlador):** O JSF possui um controlador simples, chamado de *FacesServlet* que age como um despachante, controlando o fluxo de navegação e despachando requisições para as páginas JSF apropriadas. Na Figura 5 é exemplificada a arquitetura MVC do JSF.

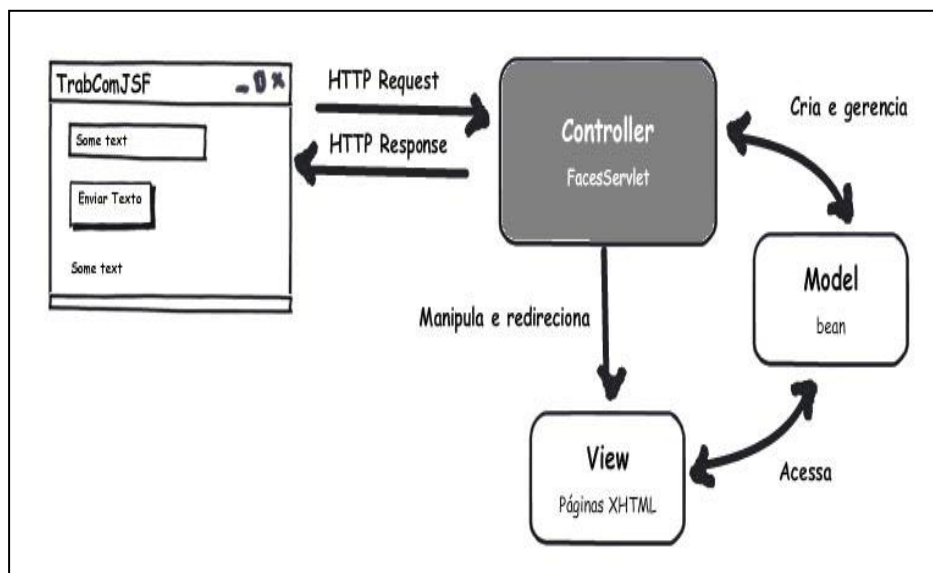
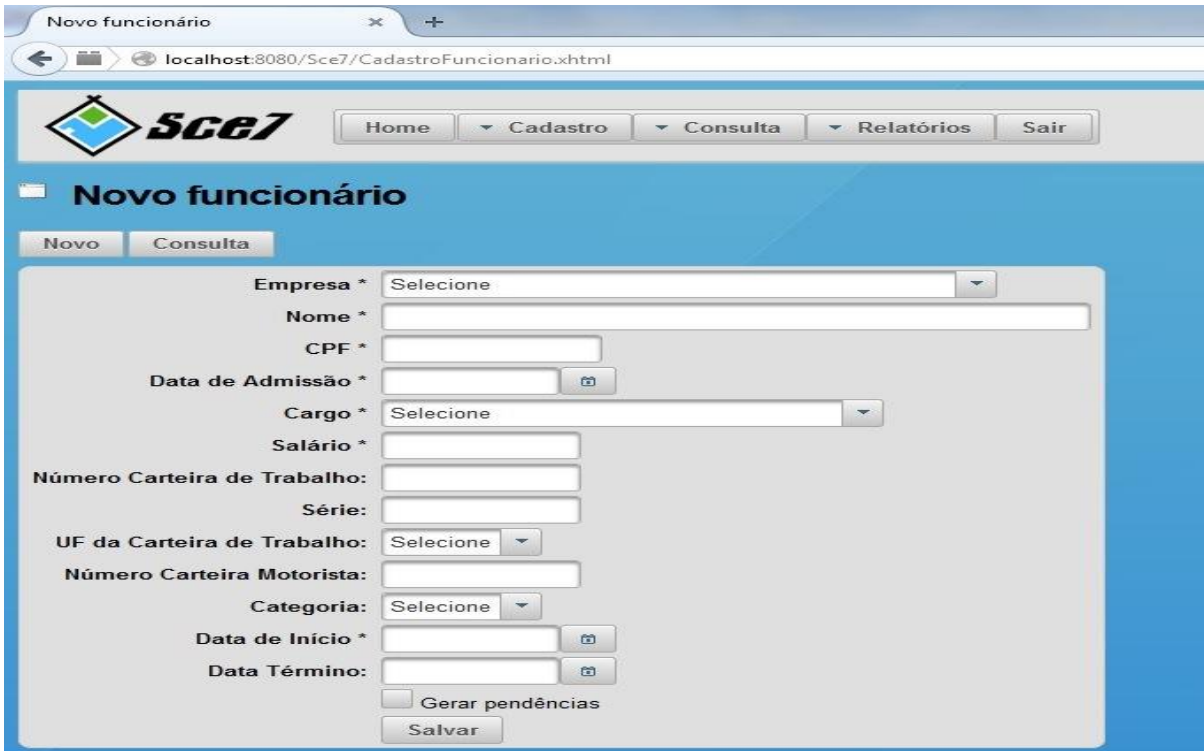


Figura 5 - O MVC do JSF.
Fonte: (GONÇALVES, 2010).

O JSF foi utilizado no para desenvolver o SCE7, com isso o sistema foi desenvolvido seguindo os padrões MVC, garantindo uma maior organização das classes e dos códigos, facilitando a manutenção do software, sua consistência e fidelidade.

2.3.6 Primefaces

Primefaces é uma biblioteca de componentes para JSF completos e de fácil implementação (MATERA, 2015). No SCE7 a maioria dos componentes como exemplo `inputText`, `Calendar`, `dataTable`, entre outros, vieram da biblioteca do Primefaces, procurando deixar o sistema com um visual agradável e interativo para o usuário devido aos componentes serem mais amigáveis e de fácil compreensão. Na Figura 6, é apresentado a tela de cadastro de funcionários do SCE7 com os componentes do Primefaces.



A imagem mostra uma captura de tela de um navegador web com o endereço `localhost:8080/Sce7/CadastroFuncionario.xhtml`. O cabeçalho da página contém o logo do SCE7 e um menu de navegação com botões para Home, Cadastro, Consulta, Relatórios e Sair. O conteúdo principal da página é o formulário de "Novo funcionário", que possui abas para "Novo" e "Consulta". O formulário contém os seguintes campos e controles:

- Empresa *: `selectOneMenu` com o valor "Selecione".
- Nome *: `inputText`.
- CPF *: `inputText`.
- Data de Admissão *: `inputText` com ícone de calendário.
- Cargo *: `selectOneMenu` com o valor "Selecione".
- Salário *: `inputText`.
- Número Carteira de Trabalho: `inputText`.
- Série: `inputText`.
- UF da Carteira de Trabalho: `selectOneMenu` com o valor "Selecione".
- Número Carteira Motorista: `inputText`.
- Categoria: `selectOneMenu` com o valor "Selecione".
- Data de Início *: `inputText` com ícone de calendário.
- Data Término: `inputText` com ícone de calendário.
- Gerar pendências.
- Botão Salvar.

Figura 6 - Cadastro de Funcionários.
Fonte: Autoria Própria.

Na Figura 6 é possível observar vários componentes do Primefaces como o `menuButton Home`, o `button Novo` e o `Consulta`, o `selectOneMenu` para a seleção das empresas, entre outros.

2.3.7 UML

A UML (*Unified Modeling Language*), sua tradução é Linguagem de Modelagem Unificada. É uma linguagem visual usada para modelar software e é baseada no paradigma de orientação a objetos. É uma linguagem de propósito geral que pode ser utilizada em todos os domínios de aplicação (OMG, 2015).

Segundo Guedes (2009), a UML tem sido nos últimos anos, a linguagem padrão de modelagem adotada em nível internacional pela indústria de engenharia de software. Em função disso, existe hoje uma grande valorização e procura por profissionais que dominem essa linguagem.

UML não é linguagem de programação, e sim uma linguagem de modelagem, cujo objetivo é de auxiliar engenheiros de software e analistas a definirem as características do sistema, tais como: requisitos, comportamento, estrutura e lógica (GUEDES, 2009).

Essa linguagem foi utilizada para projetar e documentar o software SCE7 através de diagramas. Foi utilizado no trabalho o diagrama de caso de uso, o de classes, o de sequência, o de pacotes e o de atividades. Tal escolha se deu pela facilidade de documentar requisitos e pela compreensão do que será necessário para desenvolver o sistema. Além disso, como a UML utiliza paradigma de orientação a objetos, assim como o Java, é a linguagem ideal para documentar e demonstrar o comportamento do SCE7.

2.3.8 Eclipse

O Eclipse é uma plataforma de desenvolvimento de software, é baseado em Java e oferece várias ferramentas que auxilia no desenvolvimento de *softwares* tanto em Java, como também em outras linguagens tais como C, C++ e COBOL. Foi desenvolvido inicialmente pela IBM em novembro de 2001. No ano de 2004 a Eclipse *Foundation* foi criada, fundação sem fins lucrativos e com o objetivo de gerenciar a comunidade Eclipse. Atualmente essa comunidade é composta por vários desenvolvedores de *softwares* livres, são chamados de *comitters* e trabalham

nos projetos Eclipse (IBM, 2015). A Figura 7 é exibida a interface do Eclipse.

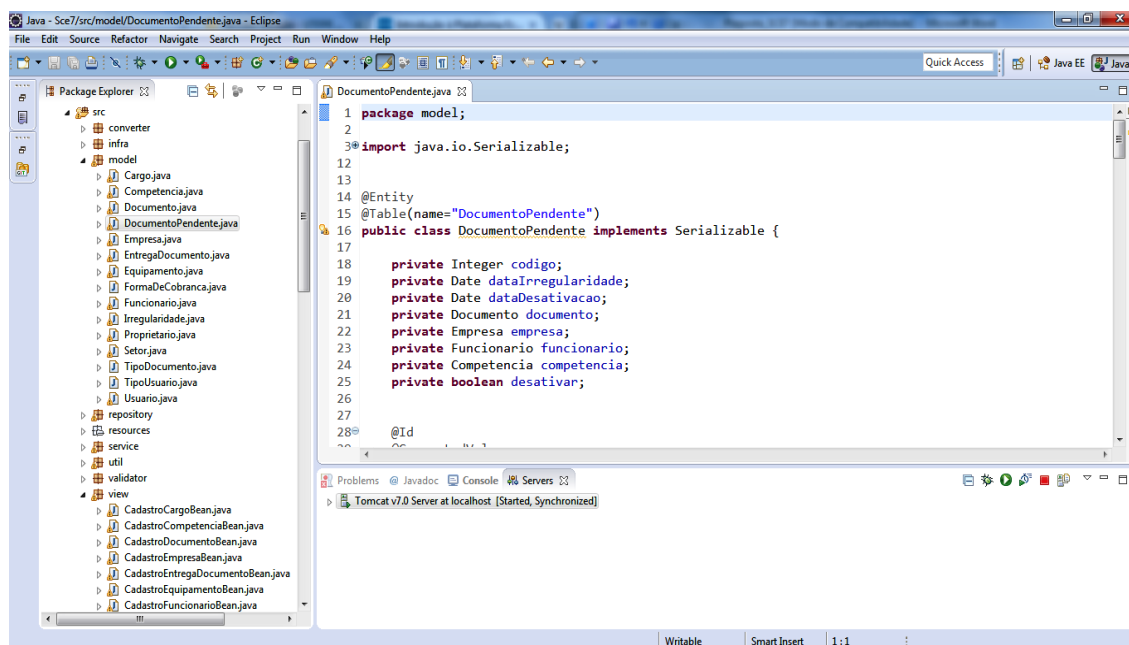


Figura 7 - Interface Eclipse.
Fonte: Autoria Própria.

No SCE7, o Eclipse foi a principal ferramenta utilizada, toda a parte de codificação com a linguagem Java e com os *frameworks* JPA, JSF, JasperReports e Hibernate, foi feita por meio do Eclipse.

2.3.9 MySQL

MySQL é o sistema gerenciador de banco de dados de código aberto mais conhecido do mundo, que traz a possibilidade de entrega econômica de aplicativos de banco de dados confiáveis, de alto desempenho e escaláveis (ORACLE, 2015).

Segundo Gonçalves (2007), o MySQL foi desenvolvido por uma empresa na Suécia que no início chamava-se TcX, posteriormente com a popularidade do MySQL, passou a se chamar MySQL AB. Seu desenvolvimento aconteceu devido a necessidade de um sistema de banco de dados que fosse extremamente rápido e

flexível. Como consequência eles criaram o MySQL, que é vagamente baseado em outro sistema de gerenciamento de banco de dados chamado de mSQL. O MySQL é rápido, flexível, confiável e é utilizado em muitos lugares por todo o mundo (GONÇALVES, 2007). Essa ferramenta foi utilizada para armazenar todas as informações do sistema, como toda a persistência é feita através do JPA juntamente com o Hibernate, esta ferramenta foi utilizada indiretamente.

2.3.10 JasperReports

O *JasperReports* é um *framework* utilizado para a geração de relatórios. É totalmente gratuito e atualmente é o mais utilizado nessa categoria. Com o *JasperReports* é possível gerar relatórios em formato de pdf, entre outros. Um aspecto importante desse *framework*, é que o *layout* do relatório é definido em um arquivo XML (K19, 2015). No SCE7 o *JasperReports* foi utilizado em conjunto com o *Ireport*, que é uma ferramenta gráfica desenvolvida para facilitar o desenvolvimento dos *layouts* no formato XML. A Figura 8 apresenta a interface gráfica do *Ireport*.

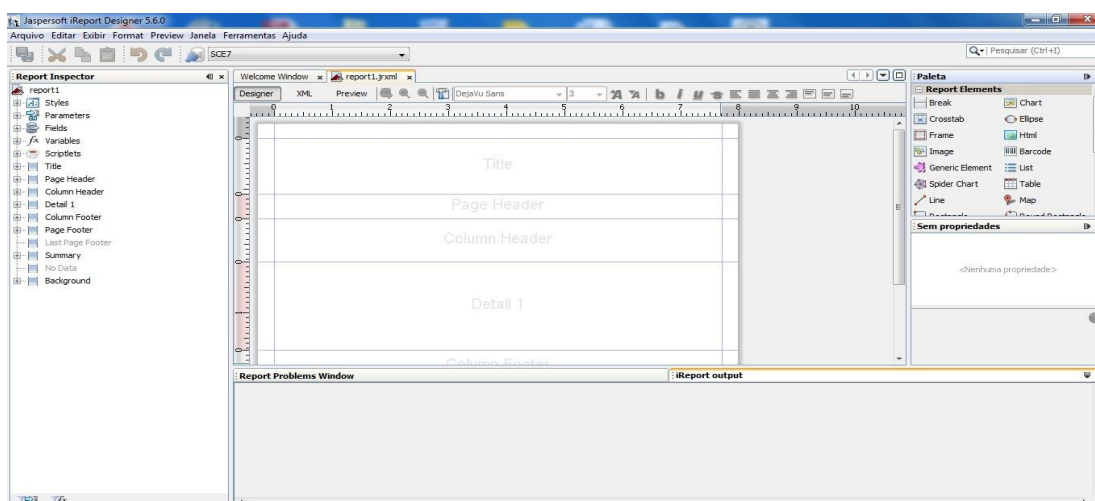


Figura 8 - Interface gráfica do Ireport.
Fonte: Autoria Própria.

2.3.11 Astah

É uma ferramenta para modelagem de *software* baseado na UML e também no paradigma de orientação a objetos. Permite a criação de qualquer diagrama, gráfico ou ilustração de forma fácil e rápida e também funciona em multi-plataforma com as características inversas, geração de código e HTML. O ASTAH também traz a possibilidade de fazer *upload* de diagramas na Web para compartilhar com outros desenvolvedores (ASTAH, 2015). Com ele foi desenvolvido os diagramas da UML, como diagrama de casos de uso e diagrama de classes. Na Figura 9 é exibida a interface do ASTAH através do diagrama de sequência.

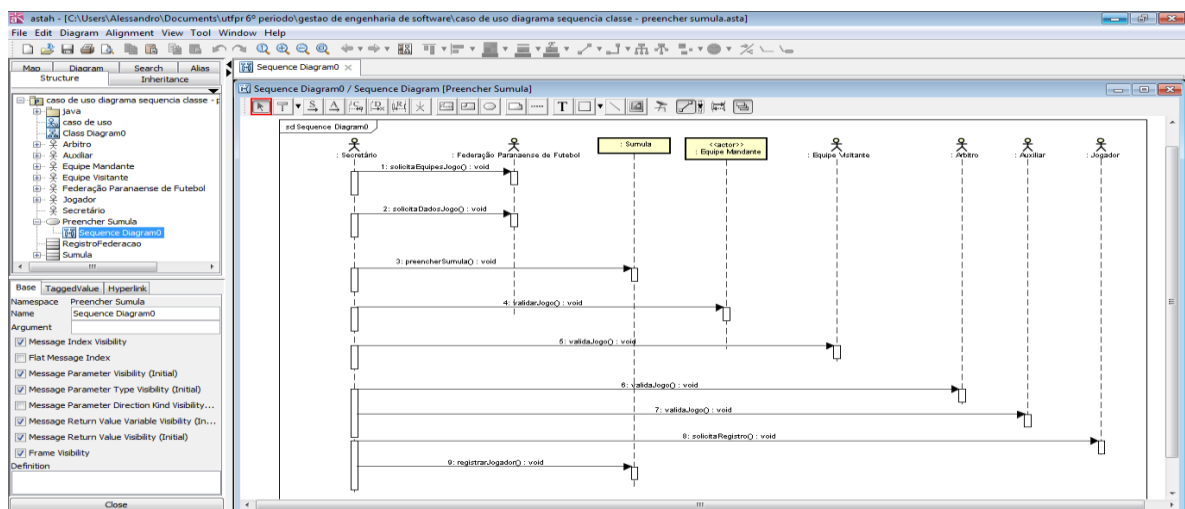


Figura 9 - Diagrama de Sequência na ferramenta ASTAH.
Fonte: Autoria Própria.

2.4 PROCESSO DE DESENVOLVIMENTO

Neste projeto foi adotada uma instanciação do método ágil Scrum. Segundo Schwaber e Sutherland (2015), o Scrum é *um framework* estrutural usado desde 1990 para a resolução de produtos complexos. Scrum não é um processo ou uma técnica para construir produtos; ao contrário disso, é um *framework* em que você pode empregar multiprocessos ou técnicas. O Scrum apresenta de forma clara

a eficácia relativa das práticas de gerenciamento e desenvolvimento de produtos, de modo que você possa melhorá-las.

Seguem as principais características do SCRUM segundo Schwaber e Sutherland (2015):

- **Backlog do produto:** O *backlog* do Produto nada mais é que uma lista ordenada de todas as necessidades do produto e é uma origem única dos requisitos para qualquer mudança a ser realizada no produto. O Backlog do Produto é de responsabilidade do *Product Owner* (dono do produto), incluindo seu conteúdo, disponibilidade e ordenação.
- **Sprint:** O coração do Scrum é a *Sprint*, a sua duração é de um mês ou menos, durante o qual será desenvolvida uma versão incremental potencialmente utilizável do produto. Dentro de cada *Sprint* tem-se uma reunião de planejamento, reuniões diárias, o desenvolvimento, uma revisão da *Sprint* e a retrospectiva. No término de uma *Sprint*, inicia-se uma nova imediatamente e o tempo da *Sprint* tem durações coerentes em todo o esforço de desenvolvimento.
- **Backlog da Sprint:** O *backlog* da *Sprint* são os itens do *backlog* do produto escolhidos para determinada *Sprint*, é a previsão do time de desenvolvimento sobre qual funcionalidade fará parte do próximo incremento e sobre o esforço necessário para entregar essa funcionalidade pronta. O *backlog* da *Sprint* torna visível todo o trabalho necessário ao Time de Desenvolvimento para atingir o objetivo da *Sprint*.
- **O Time Scrum:** O Time Scrum é formado pelo *Product Owner* que será o único responsável pelo *backlog* do produto, ele terá de fazê-lo de forma clara para que todos os membros do time entendam os itens do *backlog* e consigam por meio do mesmo saber o que será desenvolvido e em qual tempo. O Time de Desenvolvimento são todos os profissionais que realizam o objetivo de entregar uma versão usável do produto e o *Scrum Master* é o responsável pelo entendimento e aplicação do Scrum, ele trabalha como se fosse um gerente de projeto interagindo com todos os envolvidos, treinando e capacitando-os.

Os Times Scrum se auto-organizam e tem várias funções. Times auto-organizáveis buscam a melhor forma para completarem suas atividades, em vez de serem dirigidos por outros de fora do Time. Times multifuncionais têm todos os conhecimentos necessários para completar suas tarefas sem depender de outros

que não fazem parte da equipe. Times Scrum entregam produtos de forma iterativa e incremental, essas entregas incrementais de produto pronto por sua vez garantem que sempre exista uma versão funcional do produto disponível. Na Figura 10 é exemplificado o processo do Scrum citado anteriormente.

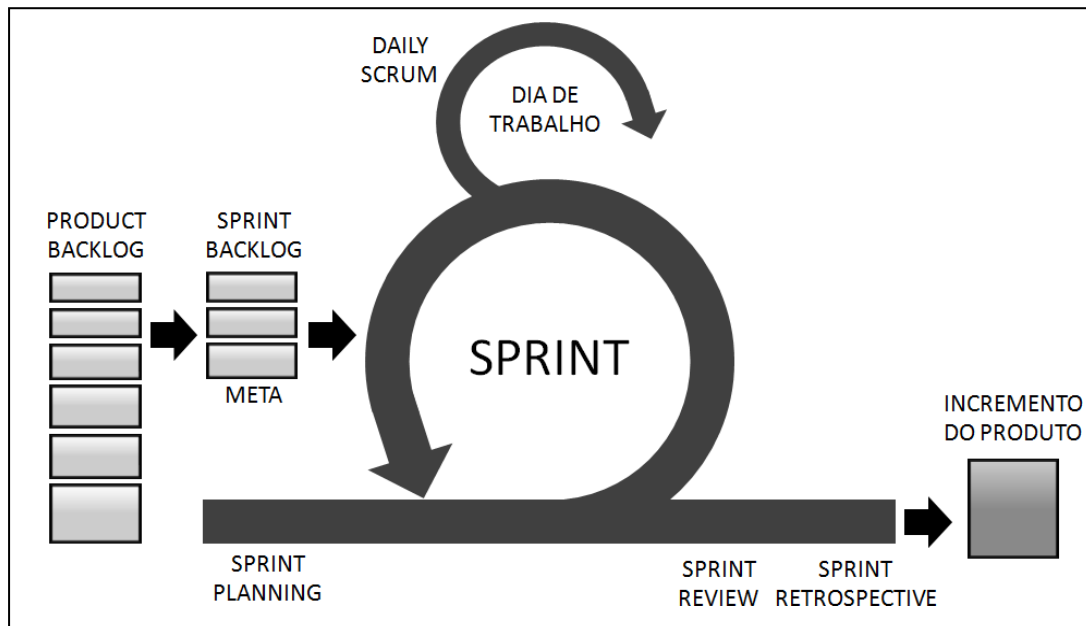


Figura 10 - Processo Scrum.
Fonte: (KNOWLEDGE21, 2015).

3. MÉTODO DE DESENVOLVIMENTO

Nesse capítulo será descrito como se deu o desenvolvimento do sistema, serão exibidos os artefatos como diagrama de casos de uso, diagrama de classes e diagrama de atividades. Além disso, são exibidos detalhes de implementação e a arquitetura do SCE7.

3.1 ARTEFATOS

Na Figura 11 é apresentado o diagrama de casos de uso do SCE7.

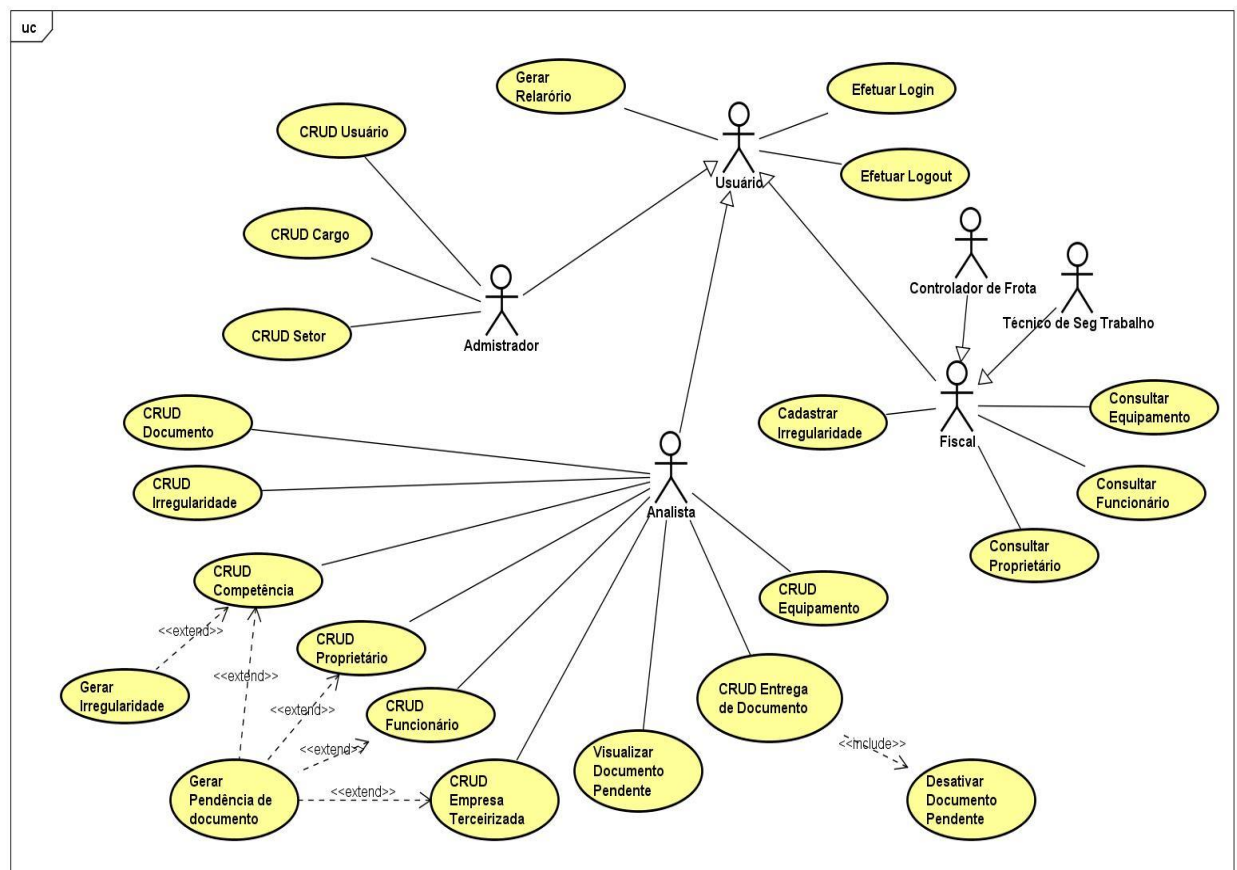


Figura 11 - Diagrama de casos de uso.
Fonte: Autoria Própria.

Segue a descrição de cada caso de uso:

- **Efetuar *Login*:** após o cadastro dos usuários do sistema pelo Administrador, o usuário efetua o acesso ao sistema através de nome de usuário e senha.
- **Efetuar *Logout*:** o usuário devidamente logado sai do sistema clicando no botão sair que por sua vez encerrará a seção.
- **Crud Empresa Terceirizada:** o analista efetua o cadastro, alteração, exclusão e consulta das empresas terceirizadas. Quando o analista cadastrar a empresa, o sistema consulta os documentos que essa empresa precisa apresentar e gera as pendências desses documentos (caso de uso: Gerar Pendência de Documento).
- **Crud Proprietário:** o analista efetua o cadastro, alteração, exclusão e consulta dos proprietários das empresas terceirizadas. Quando o analista cadastrar o proprietário, se ele for trabalhar, o sistema consulta os documentos que esse proprietário precisa apresentar e gera as pendências desses documentos (caso de uso: Gerar Pendência de Documento).
- **Crud Funcionário:** o analista efetua o cadastro, alteração, exclusão e consulta dos funcionários das empresas terceirizadas. Quando o analista cadastrar o funcionário, o sistema consulta os documentos que esse funcionário precisa apresentar e gera as pendências desses documentos (caso de uso: Gerar Pendência de Documento).
- **Crud Equipamento:** o analista efetua o cadastro, alteração, exclusão e consulta dos equipamentos das empresas terceirizadas. Os equipamentos são caminhões, tratores, guinchos, colhedoras de cana de açúcar, entre outros.
- **Crud Irregularidade:** o analista efetua o cadastro, alteração, desativação e consulta das irregularidades das empresas terceirizadas. Um exemplo desse caso de uso acontece quando o analista descobre que um funcionário está trabalhando na lavoura, porém, não consta no sistema. Então o analista lança essa irregularidade até que o proprietário traga os documentos desse funcionário. As empresas terceirizadas que possuem irregularidades tem seu pagamento bloqueado até a regularização das irregularidades.
- **Crud Documento:** o analista efetua o cadastro, alteração, exclusão e consulta dos documentos que as empresas terceirizadas precisam apresentar como exemplo carteira de motorista, programa de prevenção de riscos ambientais,

requerimento de empresário, entre outros. Os documentos podem ser referentes as empresa, aos funcionários e aos proprietários e são cobrados, no ato do cadastro, mensalmente, por data de validade ou quando acontece a demissão de um funcionário.

- **Crud Competência:** o analista efetua o cadastro, alteração, exclusão e consulta das competências, elas são usadas como referência para os documentos que são entregues mensalmente, elas representam o mês e o ano do documento. Quando o analista cadastrar a competência, o sistema consulta os documentos que as empresas terceirizadas precisam apresentar mensalmente e gera as pendências desses documentos (caso de uso: Gerar Pendência de documento). O sistema também verifica se tem algum documento vencido e gera a irregularidade do mesmo (caso de uso: Gerar Irregularidade).
- **Crud Entrega de Documento:** o analista efetua o cadastro, exclusão e consulta de entrega de documento. Em outras palavras ele irá informar ao sistema que determinada empresa terceirizada entregou um documento, o sistema automaticamente irá desativar a pendência desse documento (caso de uso: Desativar Documento Pendente).
- **Crud Usuário:** o administrador efetua o cadastro, alteração, exclusão e consulta dos usuários. O sistema conta com três tipos de usuário, o Fiscal, o Analista e o Administrador, cada um com diferentes tipos de acesso de acordo com a necessidade da empresa que utiliza o SCE7.
- **Crud Cargo:** o administrador irá cadastrar, alterar, excluir e consultar os cargos dos funcionários das empresas terceirizadas.
- **Crud Setor:** o administrador efetua o cadastro, alteração, exclusão e consulta dos setores em que a empresa tomadora de serviços terá empresas terceirizadas prestando serviço, como exemplo setor industrial ou lavoura.
- **Gerar Irregularidade:** no caso de documentos que possuem validade, quando os mesmos estiverem vencidos, o sistema gera a irregularidade deles quando o analista cadastrar uma nova competência.
- **Gerar Pendência de Documento:** através do cadastro das empresas terceirizadas, dos funcionários, das competências e dos proprietários. O sistema verifica os documentos que a empresa terceirizada deve entregar e gera a pendência desses documentos.

- **Visualizar Documento Pendente:** o analista efetua consulta dos documentos pendentes, ou seja, todos os documentos que a empresa terceirizada ainda não apresentou. As empresas terceirizadas que possuem documentos pendentes ficam com seus pagamentos bloqueados até a regularização dos mesmos.
- **Desativar Documento Pendente:** através das entregas de documentos o sistema automaticamente desativa os documentos pendentes.
- **Cadastrar Irregularidade:** o fiscal efetua o cadastro de irregularidades das empresas terceirizadas, como exemplo, funcionários que não estão registrados no sistema e estão trabalhando na lavoura. Também a casos em que os funcionários não usam os equipamentos de segurança de forma adequada.
- **Consultar Proprietário:** o fiscal efetua a consulta dos proprietários das empresas terceirizadas, a fim de verificar divergências de informação em relação ao sistema.
- **Consultar Funcionário:** o fiscal efetua a consulta dos funcionários das empresas terceirizadas, para verificar se eles estão cadastrados no sistema.
- **Consultar Equipamento:** o fiscal efetua a consulta de equipamentos das empresas terceirizadas, para verificar se os mesmos estão cadastrados no sistema.
- **Gerar Relatório:** os usuários do sistema podem gerar alguns relatórios, como listar todas as empresas terceirizadas e seus respectivos equipamentos, bem como lista de funcionários para a fiscalização na lavoura, entre outros.

Na Figura 12, é apresentado o diagrama de classes referente ao pacote *model* do SCE7. Como o sistema possui muitas classes, foram exibidas somente as classes do pacote *model*, contendo as classes entidades com todas as informações que são persistidas no banco de dados.

Na Figura 13, é apresentado o diagrama de atividades para o caso de uso gerar pendência de documento. Com ele é possível verificar em quais situações o sistema gera as pendências de documentos.

Não foi gerado o diagrama de entidade e relacionamento, pois, todas as tabelas e relacionamentos do banco de dados foram criadas automaticamente através do JPA juntamente com o Hibernate.

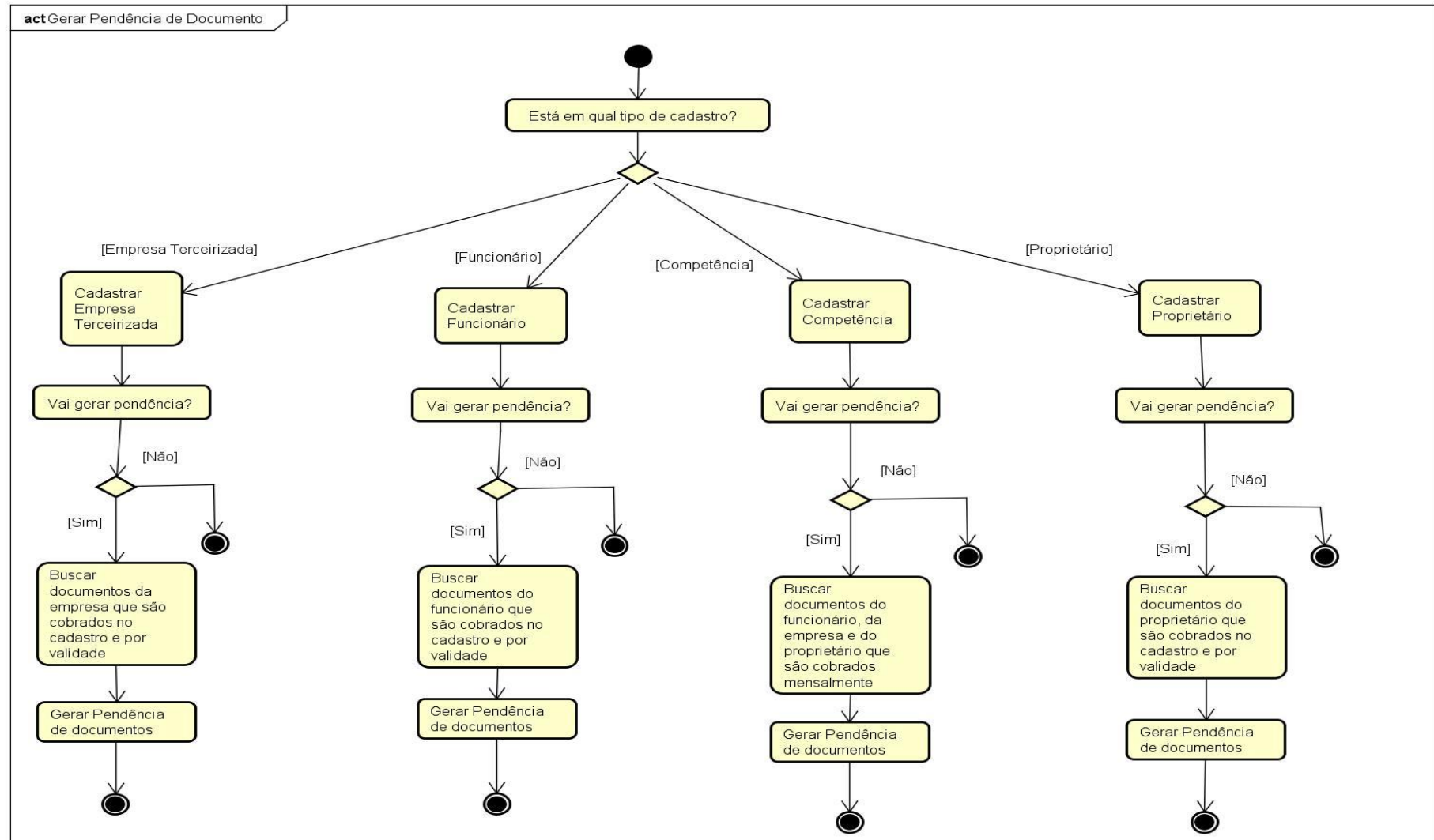


Figura 13 - Diagrama de Atividades.
Fonte: Autoria Própria.

Segue abaixo, algumas métricas para que os leitores tenham uma noção do tamanho do sistema. Para isso, foi usado um *plug-in* do Eclipse chamado de Eclipse metrics (ECLIPSE, 2015). No Quadro 2, é apresentado as métricas do SCE7.

Métrica	Valor
# Pacotes	8
# Classes	84
# Interfaces	12
# Métodos	727
# Atributos	255
# Linhas de Código	5802
# Páginas Web	34

Quadro 2 - Métricas do SCE7.
Fonte: Aatoria Própria.

3.2 PROCESSO ADOTADO

O SCE7 foi desenvolvido usando como referência o Scrum. No início deste projeto foi feito o *backlog* do produto pelo *Product Owner*, que é o autor deste trabalho, pois, o mesmo trabalha na área de atuação do sistema desenvolvido.

Por meio do *backlog* do produto foi gerado um diagrama de casos de uso geral do sistema. O *backlog* da *Sprint* era gerado com base no diagrama de casos de uso pelo Time Scrum, composto pelo autor deste trabalho, que acumula os papéis de *Product Owner* e Time Scrum e o *Scrum Master* que foi o orientador do projeto. Com o *backlog* da *Sprint* em mãos, dava-se início a *Sprint* que teve duração aproximada de duas semanas, dependendo da dificuldade era desenvolvido um ou mais casos de uso por *Sprint*, ou seja, cada caso de uso era um item do *backlog* da *Sprint*. Então a cada duas semanas era realizada uma reunião entre o Time e o *Scrum Master*, para apresentar uma demo do que foi desenvolvido na *Sprint* (*Sprint Review*) e já planejava uma nova *Sprint* (*Sprint Planning*). No Quadro 3, são apresentadas as *Sprints* do desenvolvimento do SCE7.

NOME	CASOS DE USO DESENVOLVIDOS
<i>Sprint 1</i>	Crud Cargo
	Crud Setor
	Crud Documento
	Crud Irregularidade
	Crud Funcionário
	Crud Empresa Terceirizada
	Crud Proprietário
	Crud Equipamento
	Cadastrar Irregularidade
	Consultar Equipamento
	Consultar Funcionário
	Consultar Proprietário
	Demo apresentada em 28/05/2015
	<i>Sprint 2</i>
Efetuar Login	
Efetuar Logout	
Demo apresentada em 11/06/2015	
<i>Sprint 3</i>	Crud Competência
	Gerar Irregularidade
	Gerar Pendência de documentos
	Demo apresentada em 25/06/2015
<i>Sprint 4</i>	Visualizar Documento Pendente
	Crud Entrega de documento
	Desativar Documento Pendente
	Demo Apresentada em 02/10/2015
<i>Sprint 5</i>	Gerar relatório
	Demo apresentada em 22/10/2015

Quadro 3 - Sprints realizadas no desenvolvimento do SCE7.
Fonte: Autoria Própria.

3.3 ARQUITETURA DO SOFTWARE

Na Figura 14, a arquitetura do Sistema de Controle de Empresas Terceirizadas (SCE7) é representada através do diagrama de pacotes.

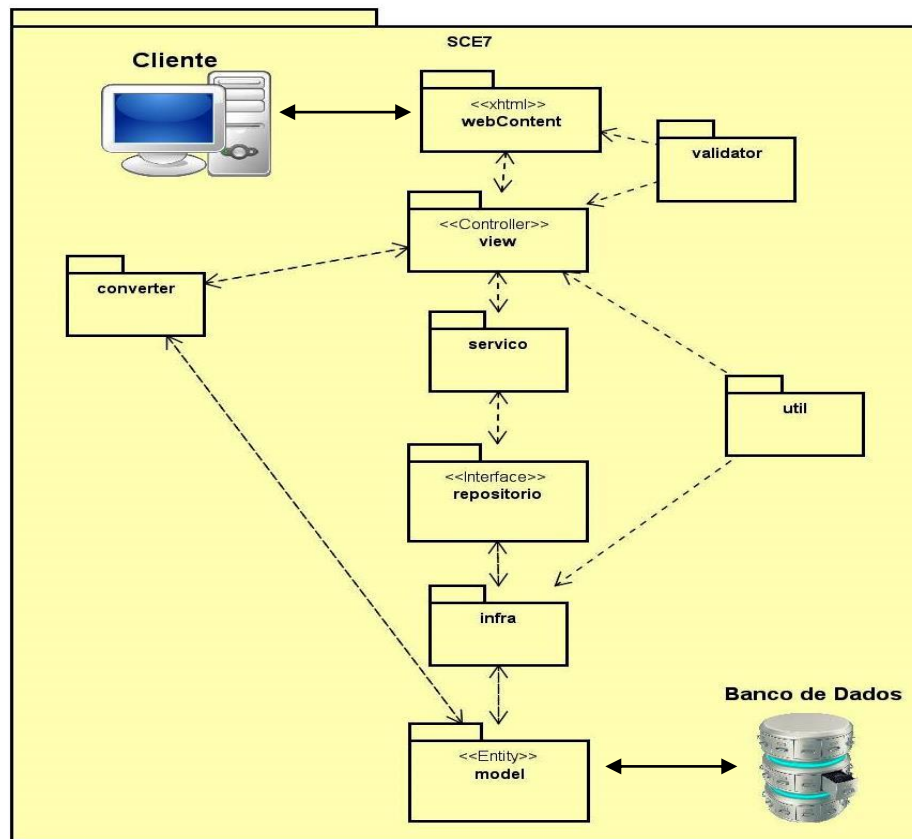


Figura 14 - Diagrama de Pacotes - Arquitetura do Software.
Fonte: Autoria Própria.

No pacote **model**, estão as classes entidades responsáveis pelas informações que são persistidas no banco de dados. Em outras palavras, tudo que o sistema precisa gravar no banco de dados estão nas classes desse pacote. As classes possuem as anotações do JPA, que efetua a conversão de classes e objetos em tabelas e campos para persistir no banco de dados informações como nome e documentos das empresas, dos funcionários e dos proprietários.

As classes do pacote **infra** contém os códigos do Hibernate; elas implementam as interfaces que estão dentro do pacote **repositorio** e mapeiam as classes que estão no pacote **model**. Por exemplo, dentro delas contém métodos que manipulam informações no banco de dados, como excluir empresa, consultar todas as empresas que estão ativas, entre outras.

O pacote **repositório** é composto por interfaces, é nelas que estão os métodos que as classes do pacote **infra** deve implementar.

O pacote **servico** contém as classes de gestão, ou seja, elas interagem com as classes do pacote **view** que são *managed beans*, recebem as informações

das mesmas e executam métodos como exemplo verificar se a informação a ser gravada já existe no banco de dados.

No pacote **view** estão as classes *managed bean*, também chamadas de *controller*, são responsáveis pela interação com as páginas XHTML do JSF, elas recebem as informações digitadas pelos usuários nas páginas e também buscam informações do banco de dados para exibir ao usuário. No pacote **webContent** contém a interface visual, ou seja, as páginas XHTML em que o usuário irá interagir, digitando informações que serão gravadas no banco, como também efetuando consultas no mesmo.

O pacote **converter** contém as classes responsáveis por efetuar a conversão de String para Objetos, pois, nos sistemas web todas as informações que vem das páginas são do tipo String. Por isso tem-se a necessidade de uma conversão, para que essas informações sejam gravadas no banco de dados.

O pacote **validator** contém classes que validam algumas informações digitadas pelo usuário nas páginas web, como exemplo validação de data futura, ou seja, o usuário não pode digitar uma data posterior a data em que ele quer gravar a informação no banco de dados.

No pacote **util** estão as classes utilitárias, que facilitam o processo de outras, como exemplo a classe hibernateUtil que é responsável por criar sessões para a gravação das informações no banco de dados.

3.4 DETALHES DE IMPLEMENTAÇÃO

Na Figura 15, é apresentado o diagrama de caso de uso cadastrar empresa. Este diagrama será usado para explicar como foi implementada tal parte do sistema.

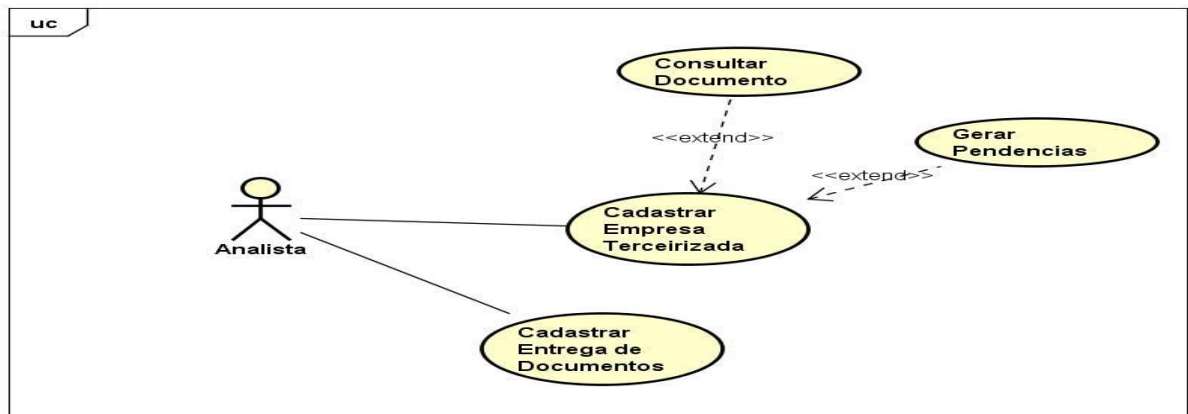


Figura 15 - Diagrama de caso de uso - cadastrar empresa.
Fonte: Autoria Própria

Para a implementação desse caso de uso, foi necessário criar as classes Empresa, Documento, DocumentoPendente e EntregaDocumento, todas no pacote *model*. Nessas classes foram adicionadas as anotações para a criação do banco através JPA juntamente com Hibernate. Na Figura 16, é exibido o diagrama de classes do pacote *model*, contendo as classes que foram usadas para implementar o caso de uso citado anteriormente.

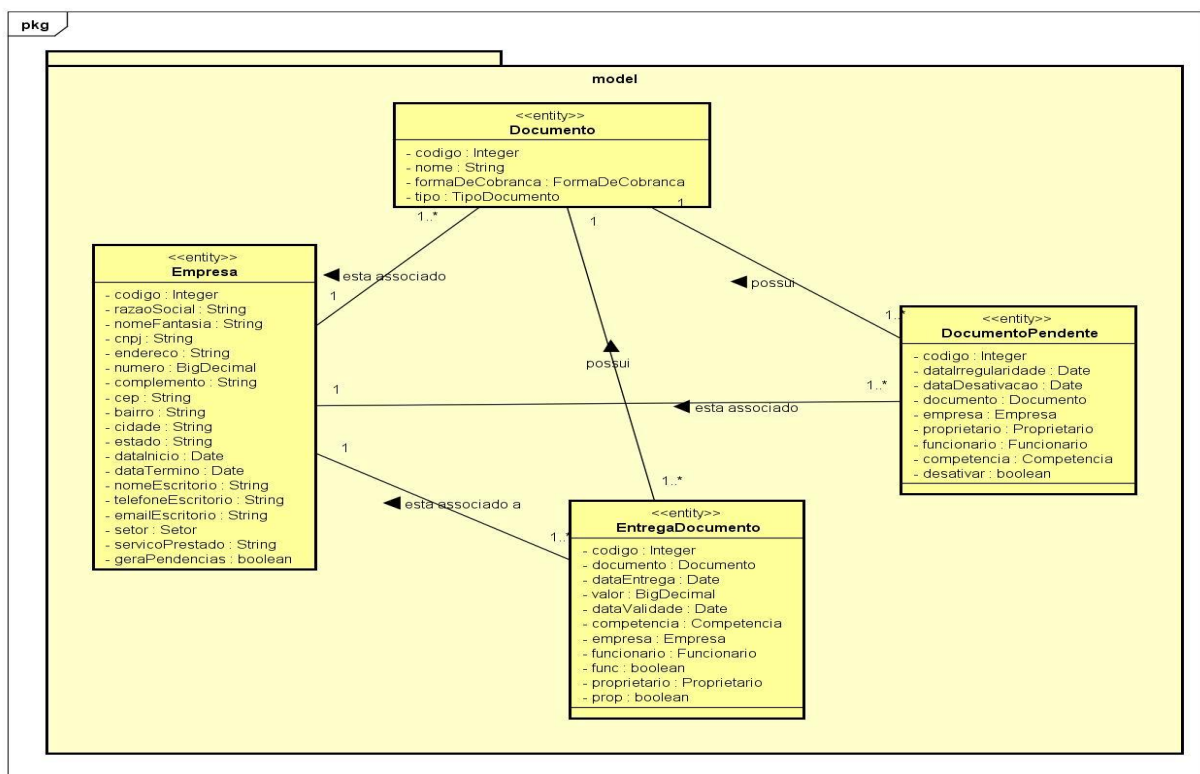


Figura 16 - Diagrama de classes do pacote model – cadastrar empresa.
Fonte: Autoria Própria.

No pacote repositório foi criado as interfaces Empresas, Documentos, DocumentosPendentes e EntregasDocumentos. Nelas contém os protótipos dos métodos que serão usados para manipular os dados do banco. Na Figura 17, é exibido o diagrama de classes interface do pacote repositório, contendo as classes interface que foram usadas para implementar o caso de uso citado anteriormente.

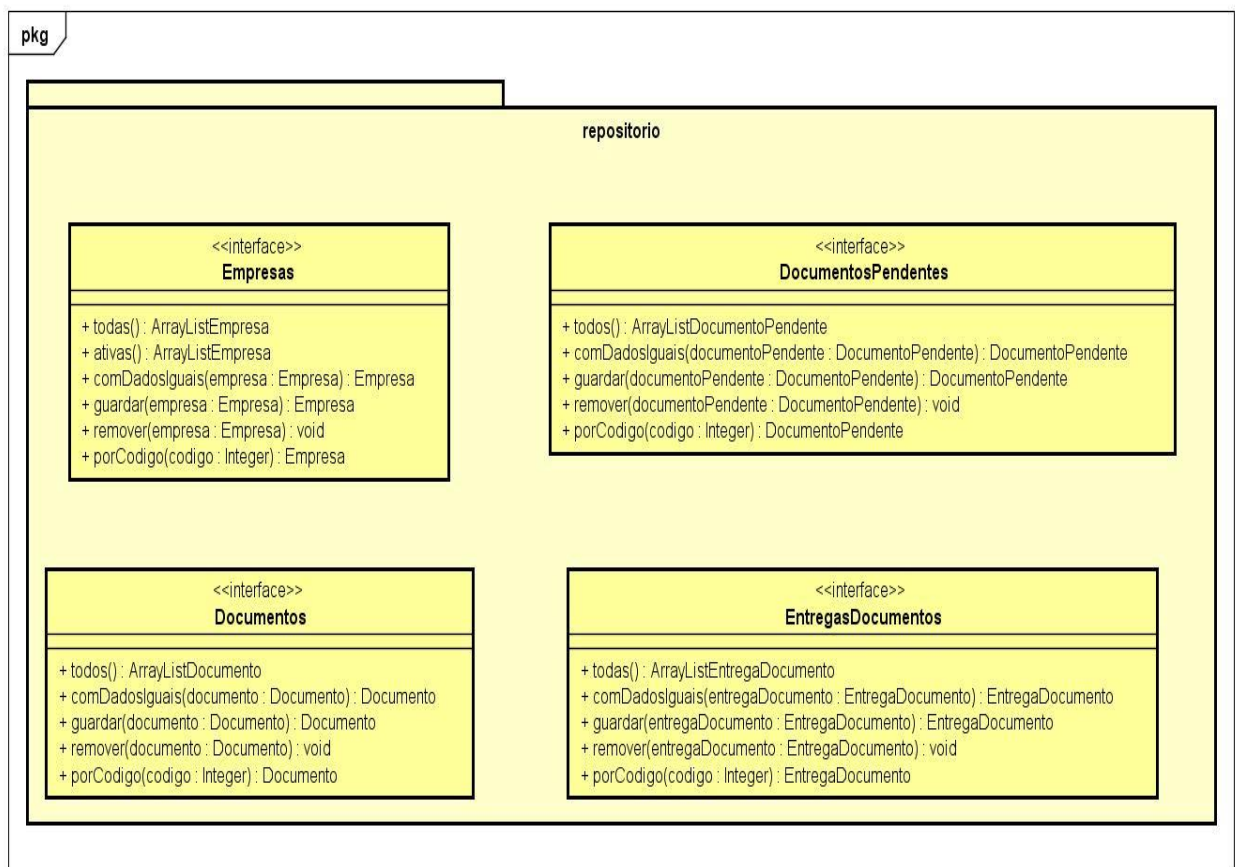


Figura 17 - Diagrama de classes do pacote repositório – cadastrar empresa.

Fonte: Autoria Própria.

No pacote infra foi criado as classes EmpresasHibernate, DocumentosHibernate, DocumentosPendentesHibernate e EntregasDocumentosHibernate. Essas classes implementam as classes interface do pacote *repositorio* e possuem os códigos hibernate que acessam o banco para fazer a manipulação de dados como cadastrar, alterar, excluir e também fazer consultas. Na Figura 18, é exibido o diagrama de classes do pacote infra, contendo as classes que foram usadas para implementar o caso cadastrar empresa.

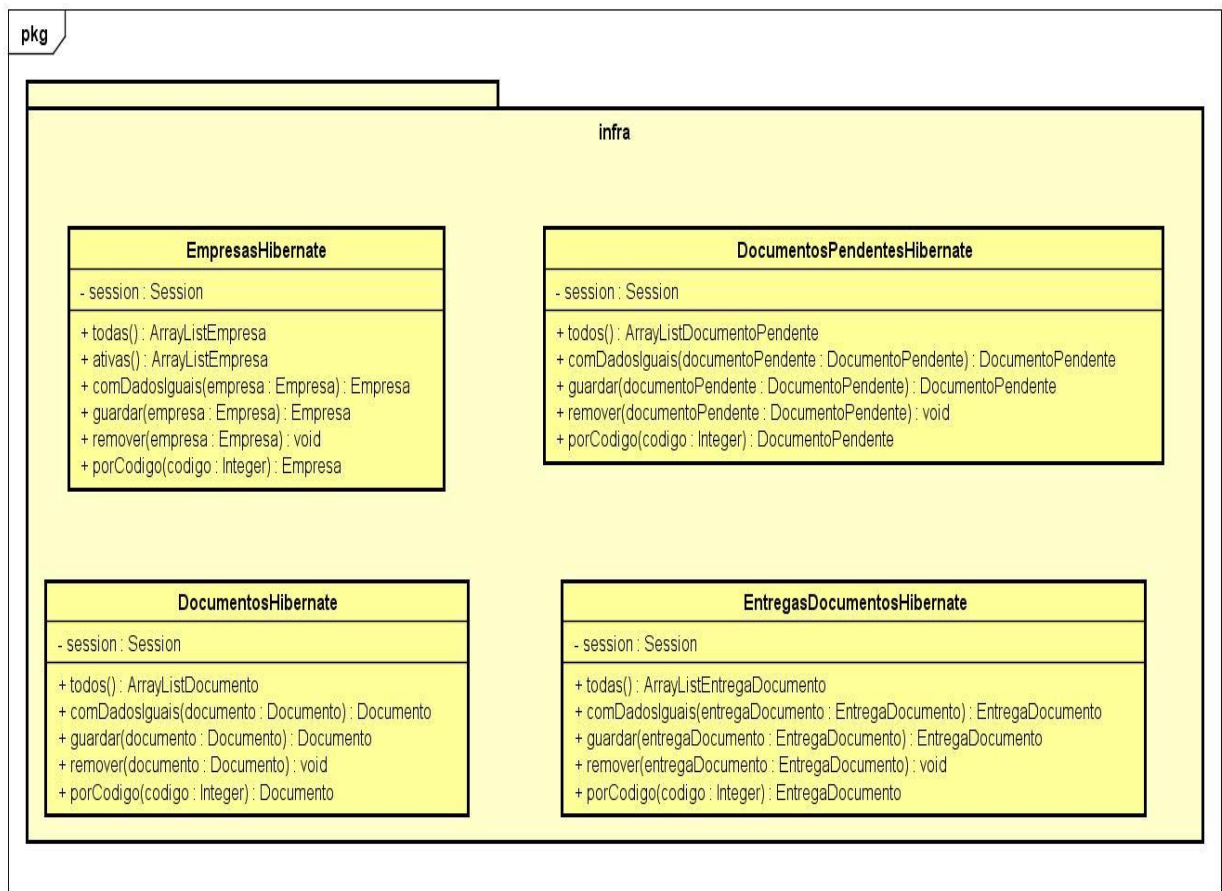


Figura 18 - Diagrama de classes do pacote infra – cadastrar empresa.
Fonte: Autoria Própria.

No pacote serviço foi criado as classes `GestaoEmpresas`, `GestaoDocumentos`, `GestaoDocumentosPendentes` e `GestaoEntregasDocumentos`. Essas classes contêm códigos de alto nível e são responsáveis por acessar as interfaces do pacote repositório e as classes do pacote infra, a fim de fazer a gestão dos dados como exemplo verificar se existe lançamentos semelhantes. Na Figura 19 é exibido o diagrama de classes do pacote serviço, contendo as classes que foram usadas para implementar o caso de uso cadastrar empresa.

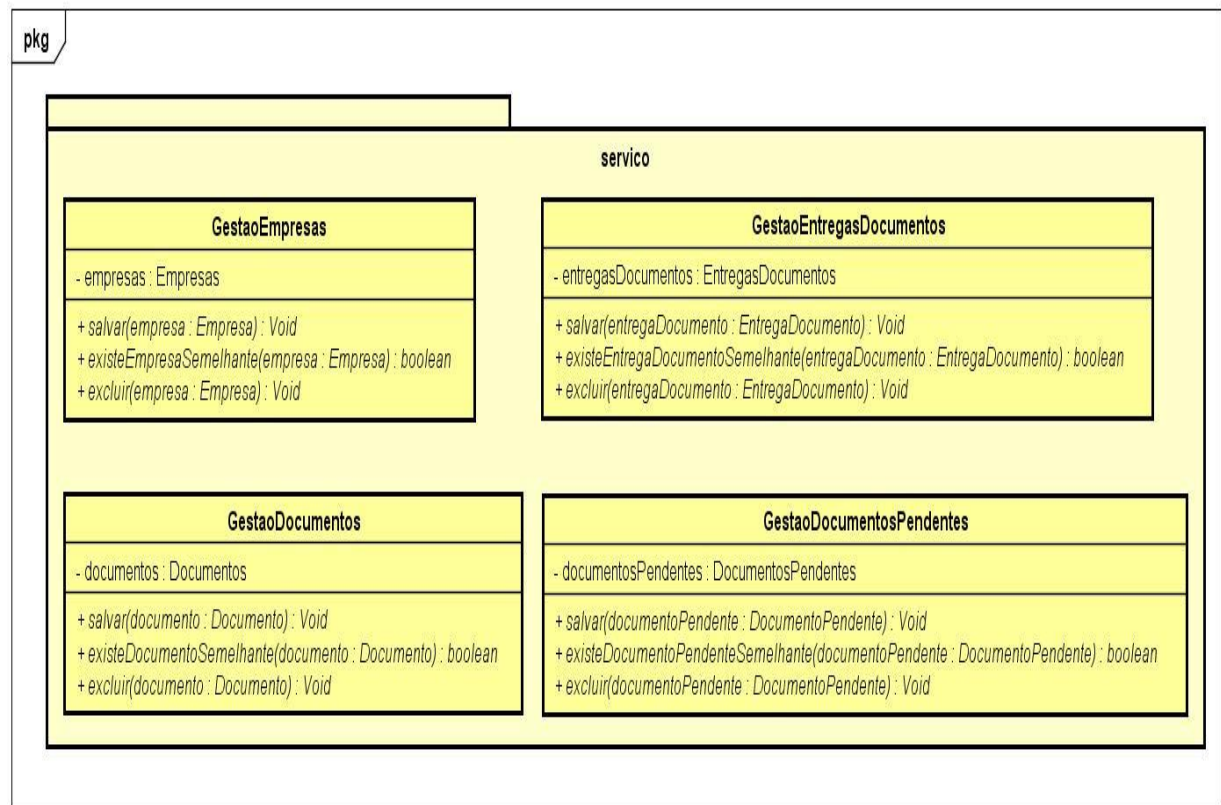


Figura 19 - Diagrama de classes do pacote servico – cadastrar empresa.
Fonte: Autoria Própria.

No pacote *view* foi criado as classes *CadastroEmpresaBean*, *ConsultaEmpresaBean*, *CadastroDocumentoBean*, *ConsultaDocumentoBean*, *ConsultaDocumentoPendenteBean*, *CadastroEntregaDocumentoBean*, *ConsultaEntregaDocumentoBean*. Essas são as classes *managedBean* ou também denominadas *controller*, são responsáveis por interagir com as páginas web do JSF e com as classes do pacote serviço. Um exemplo desse relacionamento é quando elas pegam os dados dos formulários das páginas e enviam as informações para as classes do pacote serviço para serem gravadas no banco de dados, usando assim as classes de Cadastro e também ocorre o inverso quando é necessário exibir consultas nas páginas. Na Figura 20, é exibido o diagrama de classes do pacote *view*, contendo as classes que foram usadas para implementar o caso de uso citado anteriormente.

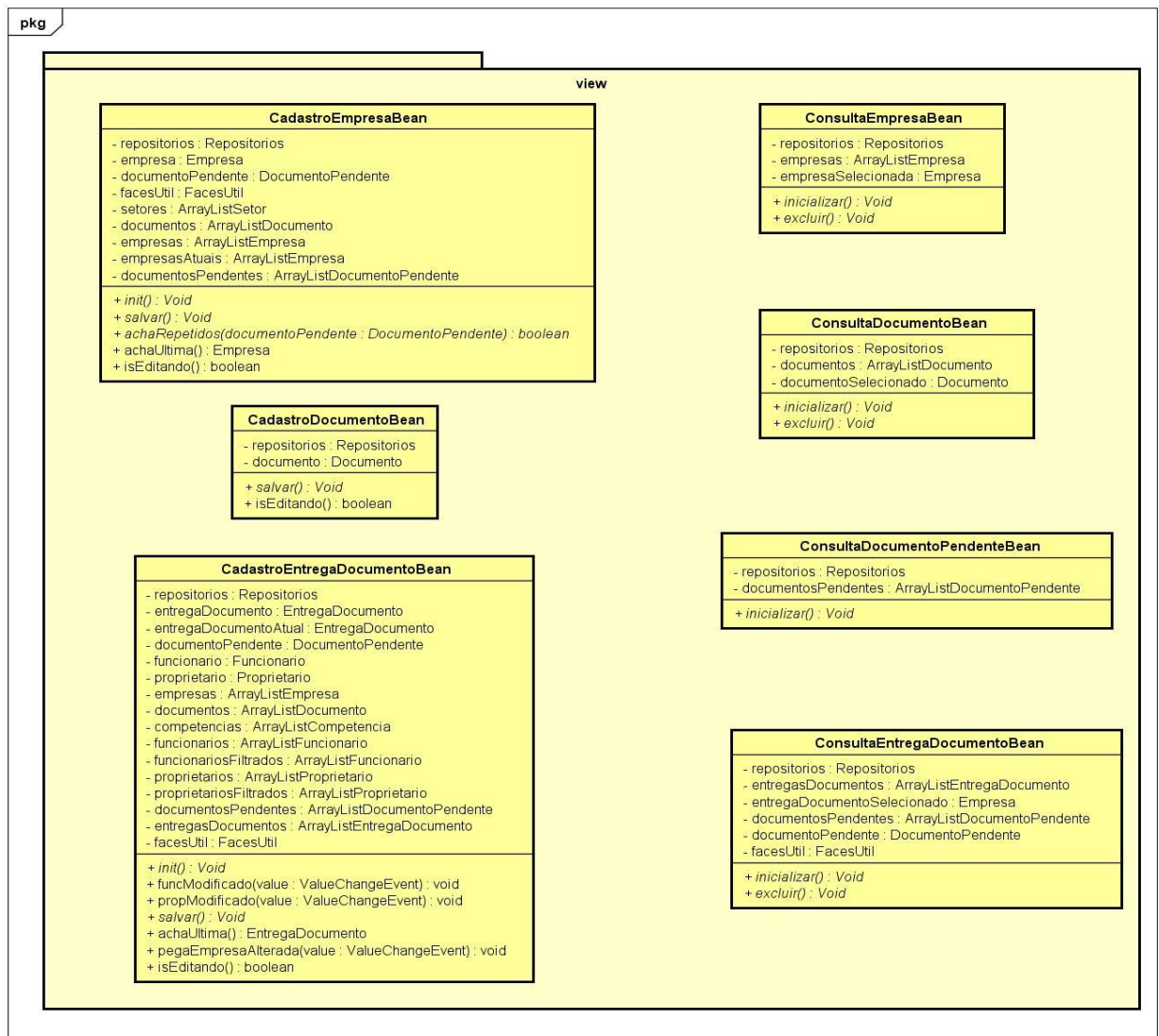


Figura 20 - Diagrama de classes do pacote view – cadastrar empresa.
Fonte: Autoria Própria.

No pacote converter foi criado as classes EmpresaConverter, DocumentoConverter e EntregaDocumentoConverter. Como os dados recebidos das páginas XHTML do JSF, vem todos como tipo String e no managedBean trabalhamos com objetos, essas classes vão converter de String para objeto e objeto para String. Na Figura 21, é exibido o diagrama de classes do pacote converter, contendo as classes que foram usadas para implementar o caso de uso cadastrar empresa.

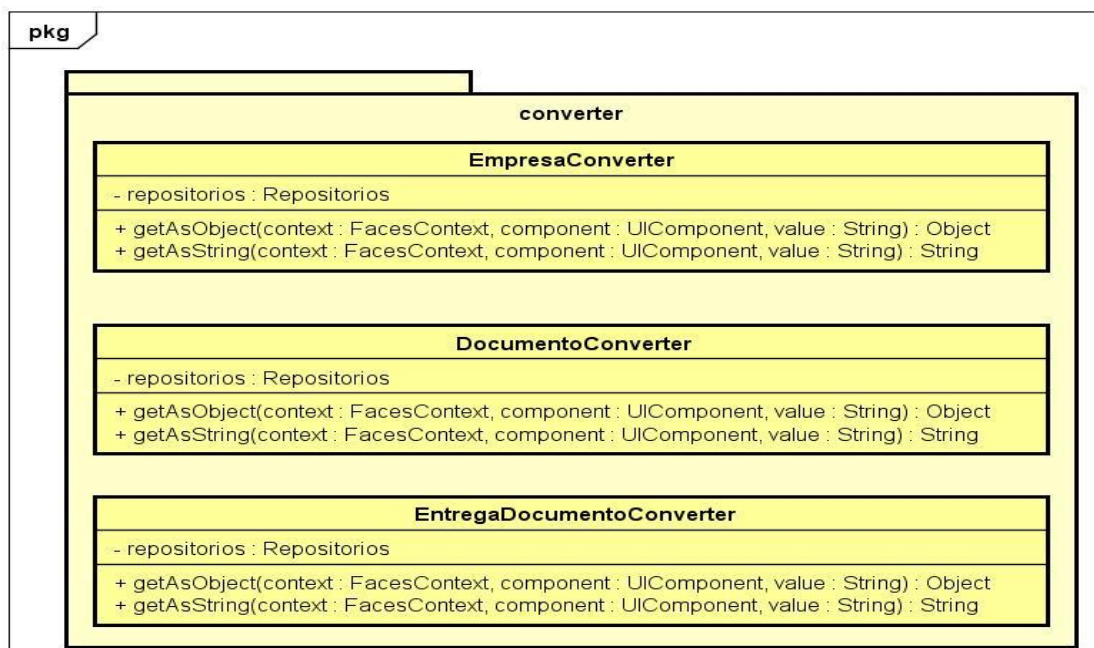


Figura 21 - Diagrama de classes do pacote converter – cadastrar empresa.
Fonte: Autoria Própria.

E por fim, na pasta WebContent estão as páginas Web CadastroEmpresa, ConsultaEmpresa, ConsultaDocumentoPendente, CadastroEntregaDocumento, ConsultaEntregaDocumento, responsáveis por interagir com o usuário e com as classes controller, a fim de persistir informações no banco de dados, bem como consultar informações no banco.

Com posse das informações citadas anteriormente, agora será explicado o caso de uso Cadastrar Empresa Terceirizada. O analista efetua o *login* no sistema, acessa a página de cadastro de empresa e preenche todos os dados da mesma. No último campo, o analista tem a opção de escolher se quer que o sistema gere as pendências de documentos ou não, supondo que o analista selecione a opção gerar pendência e clique no botão cadastrar, o mesmo irá acionar o método salvar da classe CadastroEmpresaBean que por sua vez cadastra a empresa no banco de dados. Logo após o cadastro, o sistema efetua uma consulta dos documentos que são requisitados no ato do cadastro da empresa e persiste essas pendências de documento no banco de dados.

Depois de geradas as pendências, o analista efetua o cadastro das entregas de documentos, através da página de cadastro de entrega de documento. A cada documento cadastrado é acionado o método salvar da classe

CadastroEntregaDocumentoBean, que salva o documento entregue no banco e desativa a pendência do mesmo. No fim do mês, antes da empresa tomadora de serviços efetuar o pagamento das empresas terceirizadas, o analista irá verificar se existe documentos pendentes referente a todas empresas, aquelas que estiverem com pendências não receberão seu pagamento até a regularização.

Todos os casos de uso do sistema seguem este mesmo processo. A Figura 22 apresenta o diagrama de seqüência do caso de uso para o processo geral de cadastrar empresa.

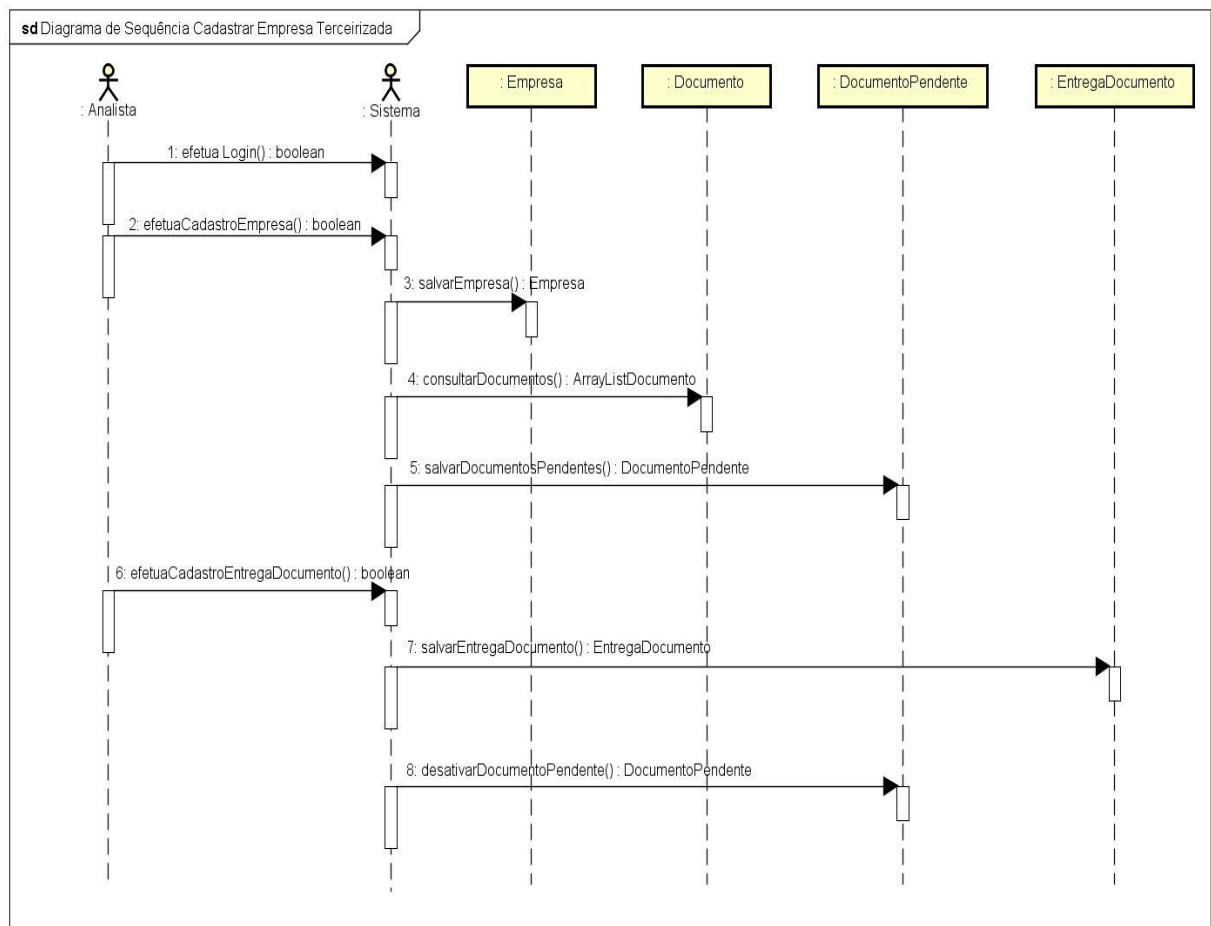


Figura 22 - Diagrama de seqüência – cadastrar empresa.
Fonte: Autoria Própria.

4. SISTEMA SCE7

Nesse capítulo são apresentadas algumas características e funcionalidades do SCE7. O sistema conta com tecnologias atuais, como o JSF 2.1 para a programação, servidor web tomcat 7.0 e o JPA juntamente com o hibernate 4.0. Todas as páginas do sistema foram desenvolvidas com componentes do primefaces 5.2, os mesmos já vem personalizados com um visual moderno e funcionalidades adicionais.

O SCE7 conta com várias validações em suas páginas de cadastro, nos campos que tem seu preenchimento obrigatório, se não estiverem preenchidos, o sistema não efetua o cadastro e informa o usuário com mensagens interativas. Em campos numéricos não é possível digitar letras, em campos de data, como exemplo data de entrega de um documento, não é possível digitar uma data futura. O sistema também possui máscaras em campos de datas, cpf, cep, cnpj, entre outros, evitando a necessidade do usuário ficar digitando barras e traços, afim de padronizar o cadastro das informações, deixando o sistema mais consistente.

Na parte de segurança, foi desenvolvida a funcionalidade de *login* através do JAAS (*Java Authentication and Authorization Service*), controlando toda entrada e saída dos usuários do sistema. Os usuários também possuem acessos diferenciados, mesmo que um usuário logado tente acessar uma página pela url de forma direta, se ele não tiver permissão para acessá-la, o sistema irá redirecioná-lo para uma página dizendo que ele não tem permissão. Na Figura 23, é apresentada a página de login.

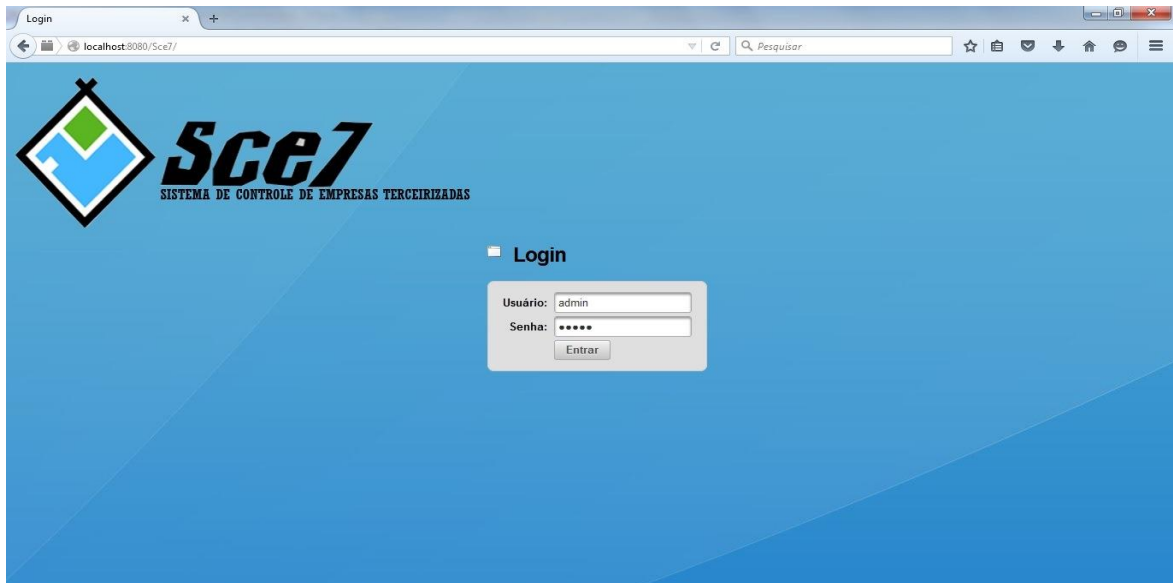


Figura 23 - Página de Login.
Fonte: Autoria Própria.

O SCE7 conta com três tipos de usuário, o fiscal, o analista e o administrador. Cada usuário tem um tipo de permissão, de acordo com os requisitos da ACUCAR E ALCOOL BANDEIRANTES SA, empresa contratante do sistema.

Na Figura 24, é apresentada a visão do fiscal quando efetua o login e clica no menu cadastro.

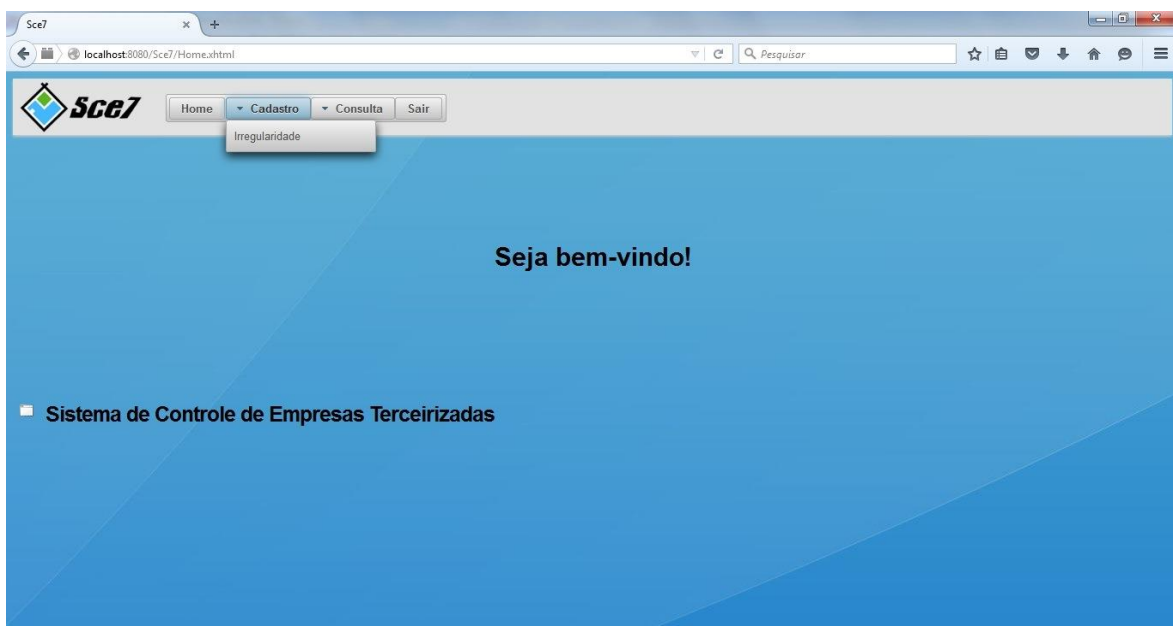


Figura 24 - Página Home, fiscal acessando o menu cadastro.
Fonte: Autoria Própria.

A Figura 24 mostra que o fiscal só pode acessar a página de cadastro de irregularidade. Na Figura 25, é apresentada a visão do fiscal quando efetua o login e clica no menu consulta.

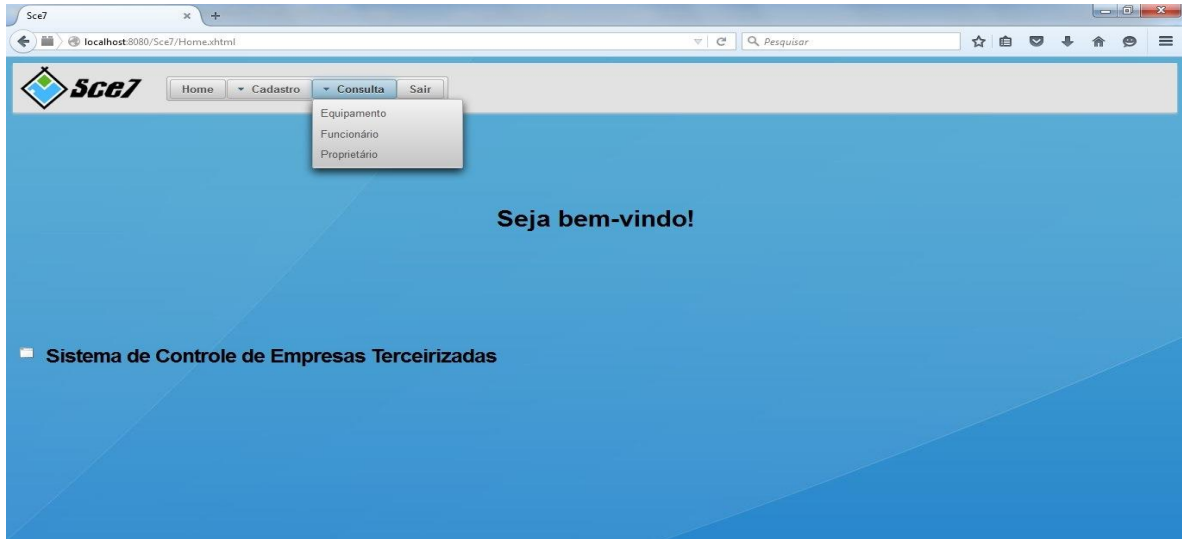


Figura 25 - Página Home, fiscal acessando o menu consulta.
Fonte: Autoria Própria.

A Figura 25 exibe o acesso do fiscal as páginas de consulta, o mesmo só pode consultar equipamentos, funcionários e proprietários das empresas terceirizadas. A Figura 26 exibe o acesso do analista as páginas de cadastro.

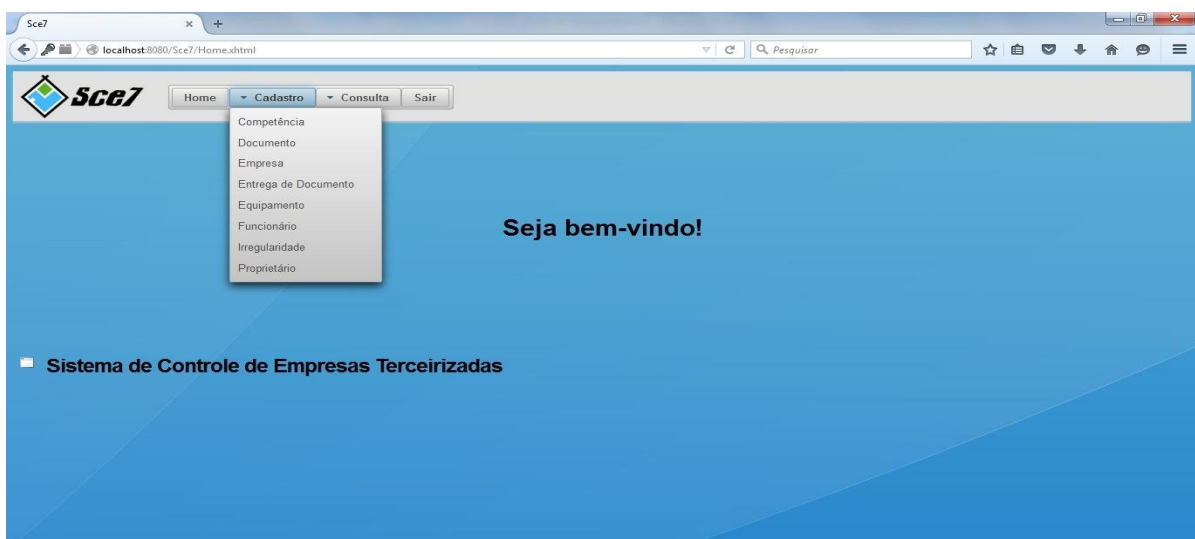


Figura 26 - Página Home, analista acessando o menu cadastro.
Fonte: Autoria Própria.

O analista é o usuário principal do SCE7, em relação às páginas de cadastro ele não tem acesso ao cadastro de usuários, cadastro de cargo, e cadastro de setor, essas páginas só podem ser acessadas pelo administrador do sistema. A Figura 27 apresenta o acesso do analista as páginas de consulta.

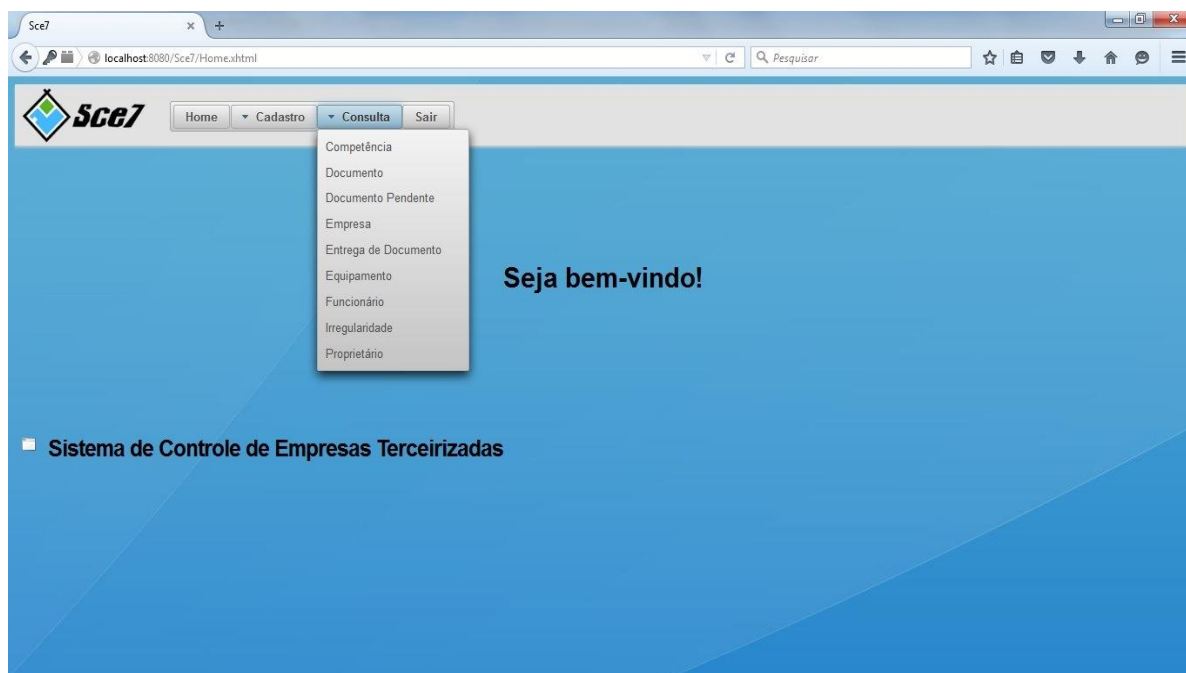


Figura 27 - Página Home, analista acessando o menu consulta.
Fonte: Autoria Própria.

O mesmo acesso que o analista tem para as páginas de cadastro, ele possui para as páginas de consulta, além disso, ele possui acesso a página de consulta de documento pendente, é esta página que o analista acessa para verificar as pendências de documento das empresas terceirizadas. Vale lembrar que quem efetua o cadastro dos documentos pendentes é o SCE7, por isso não existe uma página de cadastro de documento pendente, o sistema faz os cadastros automaticamente.

O administrador terá acesso a todas as páginas do sistema, a principal função do administrador será gerenciar os usuários, delegando as permissões de acordo com a função de cada um. Na Figura 28 é exibida a página de cadastro de empresas terceirizadas.

The image shows a web browser window with the URL 'localhost:8080/Sce7/CadastroEmpresa.xhtml'. The page title is 'Nova empresa'. The header includes the SCE7 logo and navigation links: Home, Cadastro, Consulta, and Sair. Below the header, there are two tabs: 'Novo' (selected) and 'Consulta'. The main content area contains a registration form with the following fields and values:

- Razão Social *: AGUSTINHO TRANSPORTE DE CANA
- Setor *: Selezione
- Serviço Prestado: *
- Nome Fantasia: *
- CNPJ *: 45.678.234/0001-34
- Endereço *: *
- Número: *
- Complemento: *
- Bairro *: *
- Cidade *: *
- Estado *: Selezione
- CEP: *
- Data de Início: *
- Data Término: *
- Nome do Escritório: *
- Telefone do Escritório: *
- Email do Escritório: *

At the bottom of the form, there are two checkboxes: 'Gerar pendências' and 'Salvar'. On the right side of the page, there is a vertical list of error messages, each with a red 'X' icon:

- Preencha o campo Setor.
- Preencha o campo Serviço Prestado.
- Preencha o campo Endereço.
- Preencha o campo Bairro.
- Preencha o campo Cidade.
- Preencha o campo Estado.
- Preencha o campo CEP.
- Preencha o campo Data de Início.

Figura 28 - Página Cadastro de Empresas Terceirizadas.
Fonte: Autoria Própria.

Note que na Figura 28, acima do formulário tem dois botões, o novo, e o consulta. Clicando no botão novo, o formulário será limpo, já o botão consulta irá redirecionar para a consulta de empresas. Os campos possuem máscaras, como o campo cnpj na Figura 28, os campos de data possuem calendário para facilitar a escolha da data pelo usuário. Todos os campos obrigatórios possuem o * na frente do nome do campo, se o usuário não preencher algum desses campos, são exibidas mensagens para que o usuário preencha os campos obrigatórios e os campos ficam destacados em vermelho como mostra a Figura 28. O último campo a ser preenchido é o gerar pendências, se o usuário selecioná-lo, o sistema irá gerar a pendência de todos os documentos que a empresa terceirizada precisa apresentar no ato do cadastro. Abaixo do formulário tem o botão salvar, que manda todas as informações preenchidas para o servidor.

Todas as páginas de cadastro do SCE7 possuem as mesmas características da página exibida na Figura 28. Na Figura 29, é exibida a página de consulta de documento.

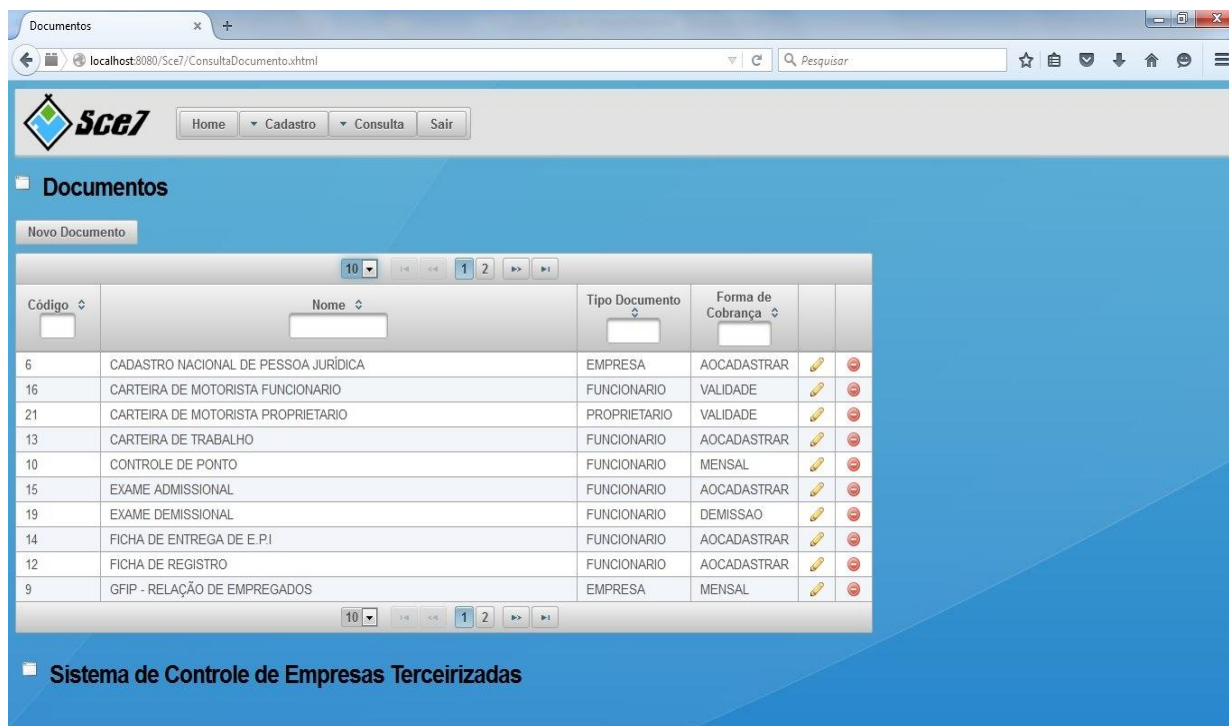


Figura 29 - Página Consulta de Documento.

Fonte: Autoria Própria.

A Figura 29 mostra a consulta dos documentos que as empresas terceirizadas precisam apresentar a empresa tomadora de serviço. Note que no canto superior esquerdo da página, existe um botão chamado novo documento, clicando nele, a página será redirecionada para o cadastro de documento. Observe que as páginas de consulta do SCE7 possuem paginação, o usuário poderá escolher a quantidade de informação que será exibida na tela, as colunas possuem filtros e se o usuário clicar no cabeçalho de uma coluna é possível ordenar as informações em crescente e decrescente.

Com base nas informações da Figura 29, é possível entender um pouco mais sobre o processo de geração de pendência de documento, que foi o processo mais complexo do sistema. Como citado anteriormente, os documentos exibidos na Figura 29, são todos aqueles exigidos pela empresa tomadora de serviço, observando a coluna tipoDocumento da figura citada anteriormente, nota-se que um documento pode ser referente a empresa, funcionário ou proprietário. No caso do proprietário, o único documento exigido é a carteira de motorista, pois existem situações em que o empresário possui um caminhão que irá transportar cana de açúcar e ele mesmo irá trabalhar com o caminhão.

Além das referências, os documentos tem diferentes tipos de cobrança,

observe na coluna formaDeCobrança que existe documento com tipo de cobrança AoCadastrar, como exemplo exame admissional do funcionário, esse tipo de documento é exigido no ato do cadastro da empresa e uma única vez. Existe a forma de cobrança Mensal, esse tipo de documento é cobrado uma vez por mês, como exemplo o holerith do funcionário, todo mês a empresa terceirizada deve apresentar o holerith dos funcionários. Existe também a forma de cobrança Validade, esses documentos são cobrados no ato do cadastro da empresa, porém, eles possuem um prazo de validade, portanto, quando os mesmos vencerem, deverão ser apresentados novos documentos, como exemplo a carteira de motorista. E para finalizar existem documentos que possuem a forma de cobrança Demissão, esses documentos são exigidos quando a empresa terceirizada efetua a demissão de um funcionário, como exemplo a rescisão do contrato de trabalho.

Para os documentos que são cobrados mensalmente, foi criado uma página de cadastro de competência, a mesma representa mês e ano, ou seja, quando a empresa efetua o pagamento de um funcionário em outubro de 2015, ela entrega o holerith do funcionário que será referente a competência Outubro de 2015. Todos os documentos que são requisitados mensalmente têm como informação sua competência.

Com posse das informações citadas anteriormente fica mais fácil entender como o SCE7 consegue gerar todas as pendências de documentos de forma correta. A Figura 30 apresenta um mapa conceitual, ilustrando como o SCE7 gera as pendências de documento.

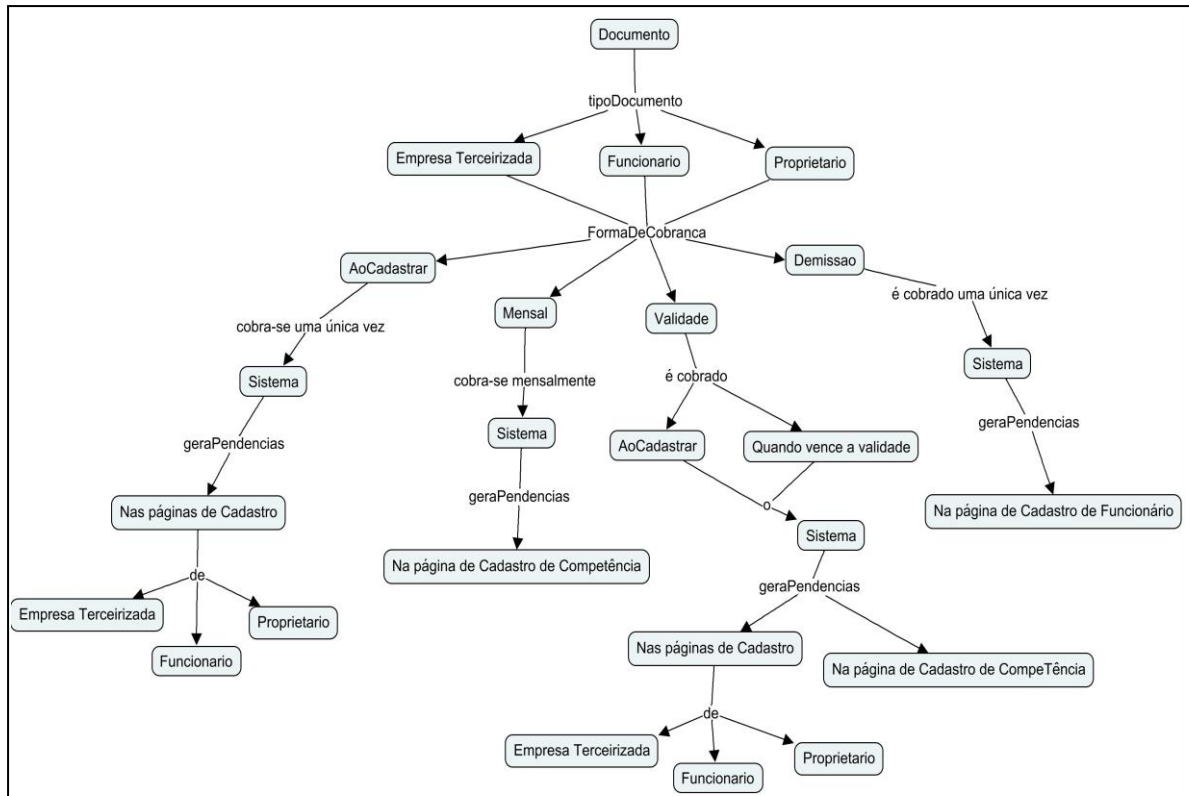


Figura 30 - Mapa Conceitual, como SCE7 gera as pendências de documento.
Fonte: Autoria Própria.

A Figura 31 apresenta o analista acessando o menu relatórios.

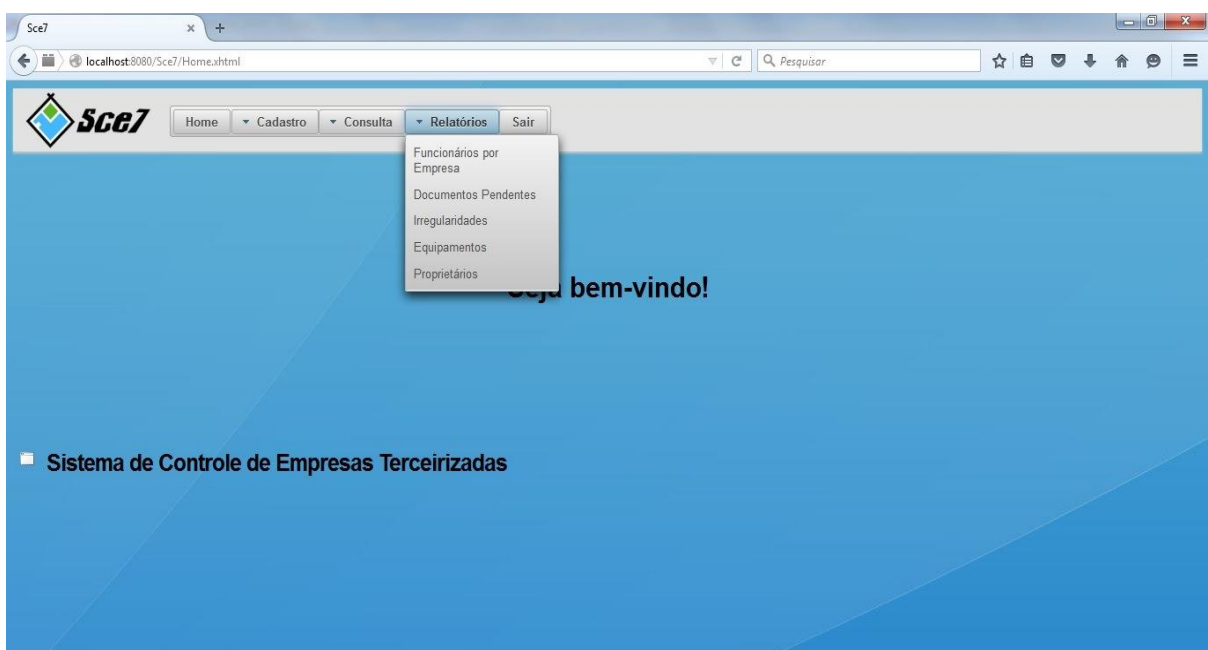
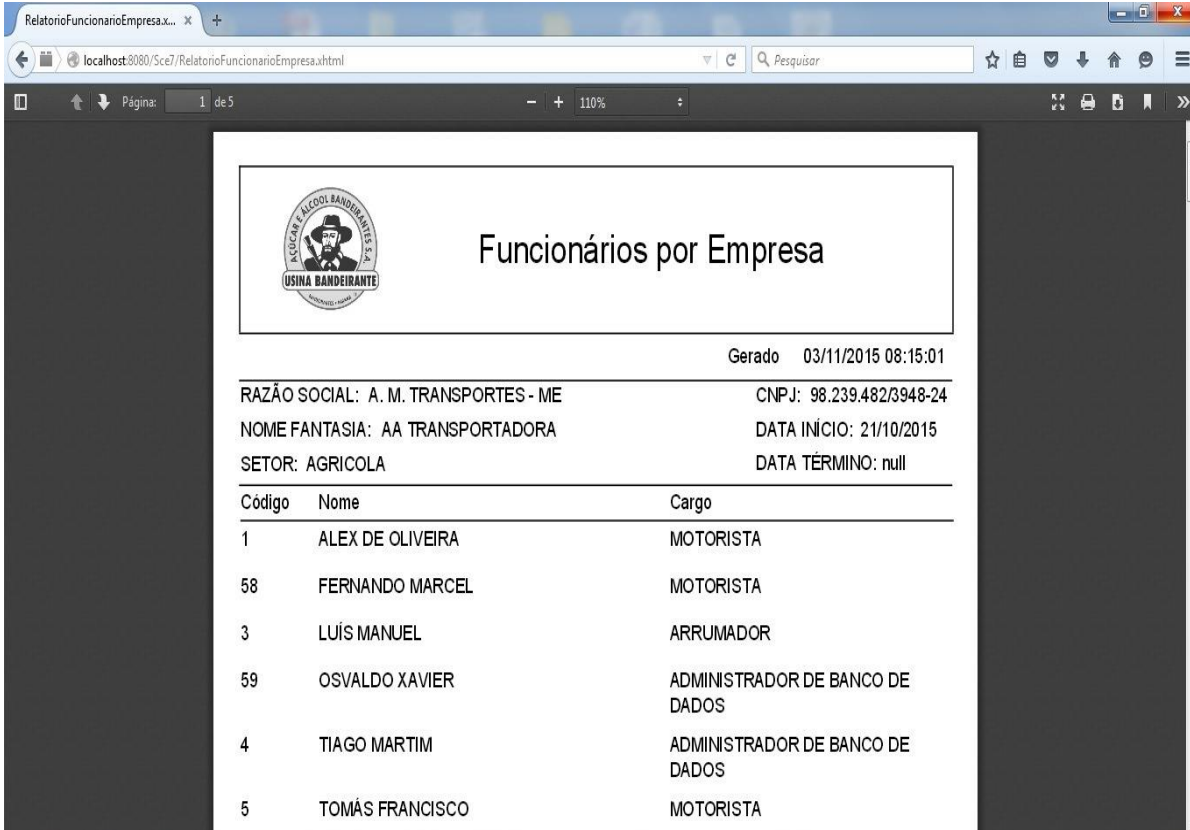


Figura 31 - Página Home, analista acessando o menu relatórios.
Fonte: Autoria Própria.

A Figura 31 exibe o menu de relatórios gerados pelo SCE7. O analista e o administrador tem acesso a todos os relatórios, já o fiscal não possui acesso ao relatório de documentos pendentes e o de irregularidades. A Figura 32 apresenta o relatório de funcionários por empresa.



RelatórioFuncionarioEmpresa.x... x

localhost:8080/Sce7/RelatorioFuncionarioEmpresa.html

Página: 1 de 5

110%

RelatórioFuncionarioEmpresa.x... x

USINA BANDEIRANTE S.A.

Funcionários por Empresa

Gerado 03/11/2015 08:15:01

RAZÃO SOCIAL: A. M. TRANSPORTES - ME CNPJ: 98.239.482/3948-24
 NOME FANTASIA: AA TRANSPORTADORA DATA INÍCIO: 21/10/2015
 SETOR: AGRICOLA DATA TÉRMINO: null

Código	Nome	Cargo
1	ALEX DE OLIVEIRA	MOTORISTA
58	FERNANDO MARCEL	MOTORISTA
3	LUÍS MANUEL	ARRUMADOR
59	OSVALDO XAVIER	ADMINISTRADOR DE BANCO DE DADOS
4	TIAGO MARTIM	ADMINISTRADOR DE BANCO DE DADOS
5	TOMÁS FRANCISCO	MOTORISTA

Figura 32 - Relatório de funcionários por empresa.
Fonte: Autoria Própria.

Na Figura 32 é gerado o relatório de funcionários por empresa. Este relatório é formado por dois relatórios, o principal exibe todas as empresas, já o secundário exibe os funcionários de cada empresa. Para isso o relatório principal envia um parâmetro com o código da empresa, com isso o relatório secundário consegue efetuar a consulta dos funcionários. É possível efetuar a impressão dos relatórios e também o download em pdf, todos os relatórios seguem o padrão do relatório exibido na Figura 32.

5. CONCLUSÃO

Nessa monografia foi apresentado o SCE7, um sistema que foi desenvolvido para informatizar o processo de controle das empresas terceirizadas que atuam na empresa Açúcar e Alcool Bandeirantes SA, trazendo consigo vários benefícios já citados na escrita desta monografia. Vale destacar o objetivo principal deste trabalho, que é a redução de custos, pois, com o SCE7, a Alcool Bandeirantes SA irá garantir que todas as empresas terceirizadas que prestam serviço a ela, paguem todos os funcionários e impostos de maneira correta, evitando o pagamento com juros desses passivos no futuro, seja ao Ministério do Trabalho ou a Justiça do Trabalho.

Foram apresentados vários artefatos que explicam de maneira detalhada todo o desenvolvimento do sistema, por meio de diagramas de casos de uso, de classes, de sequência, de atividades, de pacotes. Também foram gerados mapas conceituais para facilitar a compreensão de algumas funcionalidades. Nesta monografia também foram apresentadas as principais páginas do sistema, suas características e explicação de uso.

O SCE7 em seu estágio atual está pronto para ser implantado, o que acontecerá em breve, o sistema já passou por uma validação na ACUCAR E ALCOOL BANDEIRANTES SA, mas como a safra está em curso não será possível a implantação do sistema neste momento.

5.1 DIFICULDADES ENCONTRADAS

A principal dificuldade encontrada no desenvolvimento desse sistema, foi o processo em que o sistema tem que gerar as pendências dos documentos, pois abrange muitas classes e vários tipos de documentos que são cobrados em diversas situações. Devido o uso do JSF e das várias funcionalidades do sistema, ocorreu certa dificuldade em relação ao tempo de desenvolvimento, que se estendeu além do esperado, porém o sistema não perdeu sua qualidade.

5.2 TRABALHOS FUTUROS

Para trabalhos futuros tem-se a implantação do sistema, que já está definida para o início da próxima safra de cana de açúcar. Além disso, algumas funcionalidades do sistema serão ampliadas, na parte de entrega de documento, o sistema irá contar com um campo onde o usuário poderá fazer o *upload* de documentos. Com isso será eliminado o arquivo físico dos documentos de empresas terceirizadas, todos os documentos serão digitalizados e armazenados no banco de dados. Existe também o desejo de que o sistema, a cada mês envie um *e-mail* a todas as empresas terceirizadas informando quais são suas pendências, evitando que o analista efetue ligações para cobrar documentos das mesmas.

REFERÊNCIAS

ALLDREAMS. Disponível em: <<http://www.alldreams.com.br/informativos/13-internetsites/41-quais-sao-as-vantagens-do-aplicativo-web.html>>. Acesso em: 31 mai. 2015.

ASTAH. Disponível em: <www.astah.net/buy>. Acesso em: 02 jun. 2015.

BERNARDES, Juliana; FREIRE, Teresa. **Terceirização e responsabilidade subsidiária**. 2012. Disponível em: <<http://aplicacao.tst.jus.br/dspace/handle/1939/27256>>. Acesso em: 18 mai. 2015.

CÔRTEZ, Pedro Luiz. **Administração de sistemas de informação**. Saraiva, 2008. Disponível em: <http://scholar.google.com.br/scholar?q=Administra%C3%A7%C3%A3o+de+sistemas+de+informa%C3%A7%C3%A3o&btnG=&hl=pt-BR&as_sdt=0%2C5>. Acesso em: 01 jun. 2015.

DE AQUINO, André Wilson Avellar. **Terceirização-Responsabilidade Solidária X Responsabilidade Subsidiária**. 1998. Disponível em: <http://www.observarh.org.br/nesp/projetos/desprec/pdf/terc/aquino_andre.pdf>. Acesso em: 23 mai. 2015.

ECLIPSE. Disponível em: <<https://marketplace.eclipse.org/content/eclipse-metrics>>. Acesso em: 21 out. 2015.

GONÇALVES, Edson. **Ajax com JSF 2.0**. 2010. Disponível em: <<http://www.edsongoncalves.com.br/tag/javaserver-faces/>>. Acesso em: 01 jun. 2015.

_____. **Desenvolvendo Aplicações Web com JSP Servlets, JavaServer Faces, Hibernate, EJB 3 Persistence e Ajax**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2007. 736 p.

GUEDES, Gilleanes T. A. **UML 2: Uma abordagem prática**. São Paulo: Novatec Editora Ltda., 2009. 485 p.

HIBERNATE. Disponível em: <www.hibernate.org/orm/>. Acesso em: 30 out. 2015.

IBM. Disponível em: <<http://www.ibm.com/developerworks/br/library/os-eclipse-platform/>>. Acesso em: 03 out. 2015.

JACOBI, J.; FALLOWS, J. R. **Pro JSF e Ajax: Construindo componentes ricos para a internet**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2007. 428 p.

JAVA. Disponível em: <http://www.java.com/pt_BR/about/>. Acesso em: 14 out. 2015

K19. Disponível em: <<http://www.k19.com.br/artigos/relatorios-em-java-jasperreports-e-irepor/>>. Acesso em: 02 nov. 2015.

KNOWLEDGE21. Disponível em: <<http://www.knowledge21.com.br/tudosobreagilidade/scrum/#!/prettyPhoto>>. Acesso em: 17 mai. 2015.

MARCELINO, Paula Regina. **Afinal, o que é terceirização? Em busca de ferramentas de análise e de ação política**. Revista Pegada, v. 8, n. 2, 2012. Disponível em: <<http://revista.fct.unesp.br/index.php/pegada/article/viewArticle/1640>>. Acesso em: 23 mai. 2015.

MATERA. Disponível em: <<http://www.matera.com/br/2011/09/30/matera-utiliza-o-primefaces/>>. Acesso em: 02 nov. 2015.

OLIVEIRA, Djalma de Pinho Rebouças de. **Sistemas, organizações e métodos: uma abordagem gerencial**. 13. ed. São Paulo, 2002.

OLIVEIRA, Eva; PEREIRA, Maria João; HENRIQUES, Pedro. **Compreensão de aplicações web: O processo e as ferramentas**. 2005. Disponível em: <<https://bibliotecadigital.ipb.pt/handle/10198/577>>. Acesso em: 01 jun. 2015.

OLIVEIRA, E. C. M. **Persistência em Java com API JPA**. 2014. Disponível em: <<http://www.linhadecodigo.com.br/artigo/2525/persistencia-em-java-com-api-jpa.aspx>>. Acesso em: 01 jun. 2015.

OMG. Disponível em: <http://www.omg.org/gettingstarted/what_is_uml.htm>. Acesso em: 01 jun. 2015.

ORACLE. Disponível em: <<http://www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html>>. Acesso em: 01 jun. 2015.

_____. Disponível em: <<http://www.oracle.com/br/products/mysql/overview/index.html>>. Acesso em: 02 jun. 2015.

POLÔNIO, Wilson Alves. **Terceirização**: aspectos legais, trabalhistas e tributários. São Paulo: Atlas, 2000.

SANTOS, I. H. F. **JPA**. 2005. Disponível em: <http://www.tecgraf.puc-rio.br/~ismael/Cursos/Cidade_FPSW/aulas/Modulo2_JPA_Hibernate/JPA_Hibernate/JPA.pdf>. Acesso em: 01 jun. 2015.

SCHWABER, K.; SUTHERLAND, J. **Um guia definitivo para o Scrum**: As regras do jogo. Disponível em: <<https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide%20-%20Portuguese%20BR.pdf>>. Acesso em: 17 mai. 2015.

SISTEMASINFORMAÇÃO. Disponível em: <<http://sistemasinformacao.com/geral/definicao-de-sistema-de-informacao/>>. Acesso em: 07 jun. 2015.

TRIBUNAL SUPERIOR DO TRABALHO. Disponível em: <http://www3.tst.jus.br/jurisprudencia/Sumulas_com_indice/Sumulas_Ind_301_350.html#SUM-331>. Acesso em: 24 mai. 2015.