



UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ

CAMPUS LUIZ MENEGHEL

NATANAEL PESSOPANE FILHO

**ESTUDO DE CASO DO MÉTODO DE
DESENVOLVIMENTO SPL-OOWS**

Bandeirantes

2010

NATANAEL PESSOPANE FILHO

**ESTUDO DE CASO DO MÉTODO DE
DESENVOLVIMENTO SPL-OOWS**

Trabalho de Conclusão de Curso apresentado ao Curso de Sistemas de Informação da Universidade Estadual do Norte do Paraná, Campus Luiz Meneghel como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação, orientado pelo Prof. Msc. Bruno Miguel Nogueira de Souza

Bandeirantes

2010

NATANAEL PESSOPANE FILHO

**ESTUDO DE CASO DO MÉTODO DE
DESENVOLVIMENTO SPL-OOWS**

Trabalho de Conclusão de Curso submetido à Universidade Estadual do Norte do Paraná, Campus Luiz Meneghel, como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

COMISSÃO EXAMINADORA

Prof. Msc. Bruno M. N. de Souza

Universidade Estadual do Norte do Paraná
Campus Luiz Meneghel

Prof. Msc. Cristiane Y. H. de Castro

Universidade Estadual do Norte do Paraná
Campus Luiz Meneghel

Wellington Della Mura

Universidade Estadual do Norte do Paraná
Campus Luiz Meneghel

Bandeirantes, __ de _____ de 2010

A Deus, meus pais e amigos...

Obrigado Senhor...

AGRADECIMENTOS

Ao Prof. Msc. Bruno M. N. de Souza, que aceitou o desafio de me encaminhar nessa longa jornada.

A minha família e namorada, por me motivar e apoiar a fazer aquilo que queria.

Ao pessoal do Fórum Eleitoral, que aceitou minhas ausências quando necessário e a rapaziada que me ajudou nos momentos de dificuldade.

E acima de tudo a Deus e Nossa Senhora.

"Assim como todo reino dividido é desfeito, toda a inteligência dividida em diversos estudos se confunde e enfraquece."

Leonardo da Vinci

RESUMO

Nos dias atuais a internet se tornou um dos, senão o mais, importante meio de comunicação mundial, com isso a necessidade de se elaborar novos métodos para o desenvolvimento de aplicações Web que visam um produto final de melhor qualidade e com o menor tempo de entrega possível está crescendo muito. Com isso, este trabalho tem como objetivo desenvolver um sistema para uma locadora utilizando um novo método chamado de SPL-OOWS, que utiliza técnicas de linha de produto de software e o método de desenvolvimento de aplicações web OOWS, mostrando as vantagens de utilização desse método para desenvolvimento de softwares da mesma família de produtos.

Palavras-chave: internet, Web, SPL-OOWS, família de produtos

ABSTRACT

Nowadays the internet has become one of, if not, the most important means of global communication, thus the need to develop new methods for developing Web applications that target a final product of better quality and the shortest delivery time is growing very fast. As such, this work aims to develop a system to a video store using a new method called SPL-OOWS, that uses techniques of software product line and the method of web development OOWS, showing the advantages of using this method to software development for the same family of products.

Keyword: Internet, Web, SPL-OOWS, product family

Lista de Figuras

FIGURA 1: MODELO GENÉRICO DE GERAÇÃO DE PRODUTOS EM UMA LPS	19
FIGURA 2 : MÉTODO OOWS ADAPTADO DE FONS (2001) APUD SOUZA (2008)	20
FIGURA 3 : MODELO DE USUÁRIOS. ADAPTADO DE FONS (2003) APUD SOUZA (2008)	21
FIGURA 4 : CONTEXTO NAVEGACIONAL DE PRODUTOS DE UMA APLICAÇÃO WEB	22
FIGURA 5 : EXEMPLO DE MODELO DE NAVEGAÇÃO – FONTE – HTTP://WWW.W3.ORG/2001/SW/EUROPE/REPORTS/DEV_WORKSHOP_REPORT_8/MARIA/OOWS .MODELO.PNG	23
FIGURA 6 : ÁRVORE DE AÇÃO HIERÁRQUICA (SOUZA, 2008)	24
FIGURA 7 : MODELO SPL-OOWS - VISÃO GERAL (SOUZA, 2008)	26
FIGURA 8 : INFRA-ESTRUTURA BÁSICA (SOUZA, 2008)	27
FIGURA 9: DIAGRAMA DE CASOS DE USO DE MODELO DE USUÁRIO (SOUZA, 2008)	28
FIGURA 10 : EXEMPLO GENÉRICO DE UM CASO DE USO	29
FIGURA 11 : MODELO DE CARACTERÍSTICAS DE COMÉRCIO ELETRÔNICO	30
FIGURA 12: CASOS DE USO DO SISTEMA	36
FIGURA 13: MODELO DE CARACTERÍSTICAS DO SISTEMA	37
FIGURA 14: MODELO DE ESTADO DO CASO DE USO GERENCIAR FUNCIONÁRIO	38
FIGURA 15: MODELO DE ESTADO DO CASO DE USO GERENCIAR FILME	38
FIGURA 16: MODELO DE ESTADO DO CASO DE USO GERENCIAR CLIENTE	39
FIGURA 17: MODELO DE ESTADO DO CASO DE USO PESQUISAR FILME	40
FIGURA 18: MODELO DE ESTADO DO CASO DE USO RESERVAR FILME	40
FIGURA 19: MODELO ESTÁTICO DO SISTEMA	41
FIGURA 20: MODELO DINÂMICO INTERATIVO DO GERENCIAR CLIENTE	42
FIGURA 21: MODELO DINÂMICO INTERATIVO DO GERENCIAR FILME	42
FIGURA 22: MODELO DINÂMICO INTERATIVO DO GERENCIAR FUNCIONÁRIO	43
FIGURA 23: MODELO DINÂMICO INTERATIVO DO PESQUISAR FILME	43
FIGURA 24: MODELO DINÂMICO INTERATIVO DO RESERVAR FILME	44
FIGURA 25: MODELO GERAL DO COMÉRCIO ELETRÔNICO (SOUZA, 2008)	45
FIGURA 26: DIAGRAMA DE SEQÜÊNCIA DE GERENCIAR CLIENTE	46
FIGURA 27: DIAGRAMA DE SEQÜÊNCIA DO GERENCIAR FILME	47
FIGURA 28: DIAGRAMA DE SEQÜÊNCIA DO GERENCIAR FUNCIONÁRIO	48
FIGURA 29: DIAGRAMA DE SEQÜÊNCIA DE PESQUISAR FILME	48
FIGURA 30: DIAGRAMA DE SEQÜÊNCIA DE RESERVAR FILME	49
FIGURA 31: DIAGRAMA DE CLASSES DA LOCADORA	49
FIGURA 32: ESTRUTURA DE IMPLEMENTAÇÃO DO SISTEMA	50
FIGURA 33: MODELO DE NAVEGAÇÃO DO ADMINISTRADOR E FUNCIONÁRIO	51

FIGURA 34: MODELO DE NAVEGAÇÃO DO CLIENTE

52

FIGURA 35: MODELO DE NAVEGAÇÃO PESQUISAR FILME

52

Lista de Tabelas

TABELA 1: MODELO DE RASTREAMENTO DE VARIABILIDADES	44
TABELA 2: MAPEAMENTO DOS CONTEXTOS NAVEGACIONAIS E AS CARACTERÍSTICAS DA APLICAÇÃO	52
TABELA 3: ÁRVORE DE AÇÃO HIERÁRQUICA PARA O CONTEXTO GERENCIAR FUNCIONÁRIO	53
TABELA 4: ÁRVORE DE AÇÃO HIERÁRQUICA PARA O CONTEXTO GERENCIAR CLIENTE	54
TABELA 5: ÁRVORE DE AÇÃO HIERÁRQUICA PARA O CONTEXTO GERENCIAR FILME	55
TABELA 6: ÁRVORE DE AÇÃO HIERÁRQUICA PARA O CONTEXTO RESERVAR FILME	56
TABELA 7: ÁRVORE DE AÇÃO HIERÁRQUICA PARA O CONTEXTO PESQUISAR FILME	57

Lista de Abreviaturas

LPS	Linha de Produto de Software
OOHDM	Object Oriented Hipermedia Method
OOWS	Object Oriented Web Solution
WebML	Web Modeling Language
SPL	Software Product Line
SPL-OOWS	Software Product Line - Object Oriented Web Solution

Sumário

1 Introdução	14
2 Objetivo	16
2.1 Objetivos Gerais	16
2.2 Objetivos Específicos	16
3 Justificativa	17
4 Fundamentação Teórica	18
4.1 Linha de Produto de Software	18
4.2 Método OOWS	20
4.2.2 Modelo de Usuário	21
4.2.3 Modelo de Navegação.....	22
4.2.4 Modelo de Apresentação.....	23
4.2.5 Modelagem da Arquitetura	24
4.3 Método SPL-OOWS	25
4.3.1 Infra-Estrutura Básica.....	26
4.3.2 Modelagem Conceitual.....	29
4.3.3 Análise.....	30
4.3.4 Projeto.....	31
5 Estudo de Caso.....	34
5.1 Aplicação do Método SPL-OOWS.....	34
5.2 Domínio da Aplicação – Locadora de Filme	34
5.3 Modelagem Conceitual.....	35
5.3.1 Requisitos.....	35
5.3.2 Análise.....	37
5.3.3 Projeto.....	45
5.4 Configuração da Aplicação.....	51
6 Conclusão	58
Referências	60

1 Introdução

Nos dias atuais a palavra tempo virou sinônimo de dinheiro, com isso a Internet ganhou espaço na vida das pessoas facilitando o acesso a diversas informações além de aquisições que também podem ser feitas através deste meio. Em conjunto a isso o número de aplicações em plataforma Web esta crescendo e novos métodos para desenvolvimento deste tipo de aplicação estão sendo criados. Estes métodos visam de forma eficiente agilizar a criação de novos sistemas para esta plataforma.

Com os novos métodos surgiu à idéia de reutilização de código para se atingir um software de melhor qualidade e com maior produtividade. Buscou-se então na engenharia de software um método de reutilização, notou-se que uma técnica que tem ganhado força é Linha de Produto de Software – LPS. A LPS é definida como um conjunto de sistemas que usam softwares compartilhando um conjunto de características comuns e gerenciadas sendo desenvolvidas através de um conjunto de ativos comuns e uma forma preestabelecida que visa atender uma missão ou mercado em particular (Clements e Northrop, 2002 apud Durseki et al., 2004).

Alguns métodos de desenvolvimento Web, conhecidos no meio acadêmico são:

- OOHDM – Object Oriented Hipermídia Method (SCHWABE e ROSSI, 1998; LIMA e SCHWABE, 2003; MOURA e SCHWABE, 2004 apud Souza, 2008),
- OOWS – Object Oriented Web Solution (FONS et al., 2001; FONS et al.2003; PASTOR et al., 2001; PASTOR e INSFRAN, 1999; PELECHANO et al., 2003 apud Souza, 2008),
- WebML – Web Modeling Language (CERI et al., 2004; CERI et al., 2000 apud Souza, 2008) e UWE – UML Web based Engineering (KOCH,2001; KNAPP et al., 2004 apud Souza, 2008).

Em geral tais métodos possuem algumas etapas em comum, por exemplo, a modelagem conceitual, a modelagem navegacional e a modelagem

de apresentação. A modelagem conceitual consiste na identificação dos casos de uso, diagramas e modelagem comportamental. A modelagem navegacional demonstra a ligação entre os contextos de navegação e os usuários que têm acesso a ela. Enquanto a modelagem de apresentação consiste na parte de interação entre o usuário e a aplicação (Souza, 2008).

Porém tais métodos não são voltados para a reutilização de código, já que servem para a criação de aplicações específicas, no entanto a reutilização tem sido uma técnica de suma importância na engenharia de software para alcançar softwares de melhor qualidade e produtividade.

O trabalho propõe o desenvolvimento de uma aplicação Web para locação e venda de filmes, utilizando uma extensão do método OOWS que inclui conceitos de LPS, chamada de SPL-OOWS, criada por Souza (2008).

A estrutura do trabalho está dividida da seguinte forma: o trabalho possui seis seções. A Seção 2: apresenta os objetivos do trabalho, sendo esta dividida em duas subseções, seção 2.1 mostra os objetivos gerais e seção 2.2 mostra o objetivo específico do sistema. A seção 3 mostra a justificativa que levou ao desenvolvimento deste trabalho. A seção 4 apresenta a Fundamentação Teórica do trabalho e esta dividida em três subseções, a 4.1 aborda sobre Linha de Produto de software, a seção 4.2 apresenta o método de desenvolvimento de aplicações Web OOWS, esta seção é dividida em cinco subseções. A seção 4.2.1 fala sobre a Modelagem Conceitual de um aplicativo a ser desenvolvido. A 4.2.2 fala sobre Modelo de Usuário, onde apresenta como é feita a divisão dos usuários e a classificação de cada. Seção 4.2.3 Modelo de Navegação fala sobre como é montada o esquema navegacional do site, o que o usuário pode “ver” da aplicação. Seção 4.2.4 Modelo de Apresentação que deve cumprir o que está sendo apresentado. 4.2.5 Modelagem da Arquitetura aborda sobre as definições de cada camada do modelo de arquitetura. A subseção 4.3 apresenta o método SPL-OOWS e está dividida em 4 subseções. 4.3.1 Infra-estrutura básica de uma aplicação, 4.3.2 Modelagem conceitual do método, 4.3.3 Análise do projeto e 4.3.4 Projeto. A seção. A seção 5 apresenta o estudo de caso realizado no trabalho e a seção 6 apresenta as conclusões obtidas após o estudo.

2 Objetivo

2.1 Objetivos Gerais

Demonstrar o comportamento do método SPL-OOWS que utiliza técnicas de Linha de Produto de Software em conjunto com um método de desenvolvimento de aplicações web, tentando demonstrar uma possível melhora na qualidade do produto final através de técnicas de reutilização de código e artefatos.

2.2 Objetivos Específicos

Desenvolvimento de uma aplicação web para locação e venda de filmes utilizando o método SPL-OOWS a fim de verificar se o método atende as expectativas de quem irá utilizá-lo.

Primeiramente será feita a leitura e entendimento do método SPL-OOWS, para posteriormente fazer-se a diagramação UML do sistema utilizando o software astah community para uma avaliação do método de desenvolvimento.

3 Justificativa

Com a escassez de tempo do mundo atual e com a grande concorrência entre as empresas de desenvolvimento, há a necessidade de um diferencial de tempo no desenvolvimento de uma aplicação e de um produto final de melhor qualidade.

O método SPL-OOWS será utilizado no trabalho por apresentar características que melhoram o desenvolvimento da aplicação produzindo um produto final de melhor qualidade, através de técnicas de padronização de etapas e de infra-estrutura.

4 Fundamentação Teórica

4.1 Linha de Produto de Software

A LPS é definida como um conjunto de sistemas que usam softwares compartilhando um conjunto de características comuns e gerenciadas sendo desenvolvidas através de um conjunto de ativos comuns e uma forma preestabelecida que visa atender uma missão ou mercado em particular (Clements e Northrop, 2002 apud Durseki et al., 2004).

A LPS também conhecida como famílias de produtos de software visa estabelecer o reuso sistemático de software, ou seja, reuso planejado de componentes, fazendo a reutilização de ativos base dentre os produtos (Greenfield et al., 2004 apud Teixeira, 2007), além da customização em massa que entende-se por produção de produtos padronizados em larga escala às necessidades dos clientes (Davis, 1997 apud Teixeira, 2007).

O objetivo da LPS é minimizar os custos de desenvolvimento e manutenção de softwares que pertencem a um mesmo domínio. Para que o desenvolvimento em massa com baixo custo possa ser feito há a necessidade de uma base comum, denominada plataforma ou arquitetura da LPS. A arquitetura deve prever os possíveis produtos gerados a partir de uma linha em comum, portanto ela deve não só as características em comum dos produtos, mas também as possíveis mudanças das características além das opcionais. Esse método possui duas etapas: (i) Engenharia de Domínio e (ii) Engenharia de Aplicação. A Engenharia de Domínio também conhecida como a etapa de desenvolvimento da linha de produtos, faz a modelagem da arquitetura que servirá de base á LPS, além da definição das características comuns e variáveis da LPS. É composta pelos artefatos gerados, também chamados de ativos de base, por exemplo, documentos de requisitos, bibliotecas de código, casos de teste. Estes ativos podem ser comuns a todos os produtos da LPS ou podem ser variáveis ou opcionais de alguns (Teixeira, 2007).

A Engenharia de Aplicação, conhecida também como desenvolvimento de produto, é responsável pela criação dos produtos específicos a partir da LPS estabelecida na etapa de Engenharia de Domínio. Tal processo de criação de um produto a partir de uma LPS é também conhecido como instanciação ou derivação do produto. A figura 1 mostra o processo de geração de produtos usando ativos base e configurações do produto. Esta configuração, também chamada de modelo de decisão, consiste na escolha dos ativos opcionais e variáveis, que irão produzir o produto em conjunto com os ativos comuns a todos os outros produtos.

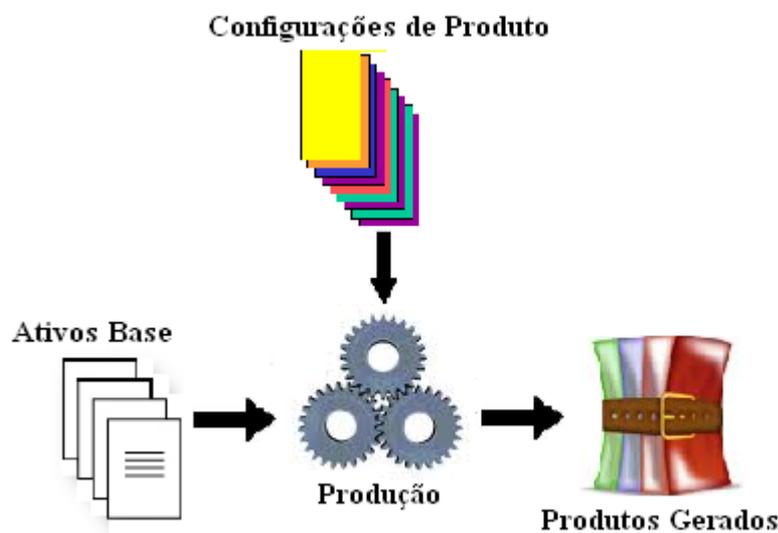


Figura 1: Modelo genérico de geração de produtos em uma LPS

Alguns dos benefícios que a LPS pode trazer são: redução dos custos de desenvolvimento; melhora na qualidade pelo reuso; redução de tempo para desenvolvimento e comercialização; redução de esforço na manutenção.

Tal método é voltado para desenvolvimento em larga escala, o desenvolvimento se dá através do reconhecimento das semelhanças e variabilidades de um conjunto de software em um domínio produzindo então o núcleo de artefatos comuns ao software, de tal forma que os produtos de software possam ser desenvolvidos a partir do núcleo podendo adicionar ou não artefatos que implementem as partes variáveis dos produtos. Esta técnica permite que grande parte dos requisitos e da arquitetura seja reusada (Queiroz, 2008 apud Lobo e Rubira, 2009).

4.2 Método OOWS

O método OOWS é uma extensão de um método orientado a objeto baseado no OO-Method (PASTOR e INSFRAN, 2001 apud Souza, 2008). O processo OOWS possui 2 (duas) fases distintas, a especificação do sistema e o desenvolvimento da solução. A especificação do sistema é dividida nas seguintes etapas: levantamento de requisitos funcionais e modelagem conceitual. O levantamento de requisitos possui 3 (três) modelos: dinâmico, semântico e funcional. Na fase de desenvolvimento, busca-se a geração de elementos de software que apresentam uma solução tecnológica para o sistema, é nesta fase que é feita a definição da arquitetura do sistema e sua implementação. A arquitetura adota 3 (três) camadas: apresentação, aplicação e persistência. A figura 2 esquematiza o método OOWS.

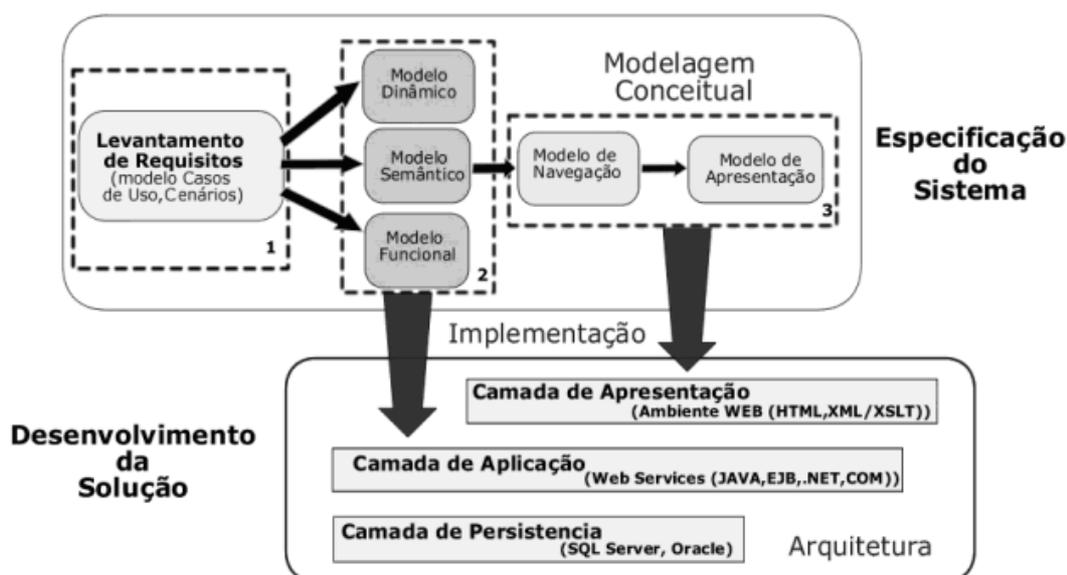


Figura 2 : Método OOWS adaptado de Fons (2001) apud Souza (2008)

4.2.1 Modelo Conceitual

A modelagem conceitual consiste no levantamento dos requisitos para a aplicação Web, gerando a especificação do sistema. As especificações podem ser classificadas através da funcionalidade, navegabilidade e especificidade das aplicações Web. Os diagramas representativos da modelagem conceitual são: diagrama de casos de uso, de classes, de estado e especificação textual

definindo os estados. Na modelagem da apresentação e navegação, a partir do diagrama de classes e dos requisitos de navegação é possível criar o modelo navegacional. Enquanto a especificação de requisitos de apresentação é feita utilizando um modelo de apresentação baseado no modelo navegacional.

O método OOWS utiliza três métodos do OO-Method conforme descritos abaixo.

4.2.2 Modelo de Usuário

O modelo de usuário especifica que tipo de usuário pode interagir com o sistema e qual a sua visão da aplicação. Para isso há uma representação hierárquica dos atores, existem três tipos de atores (usuários): anônimo, registrado e genérico, de acordo com a figura 3.

Um usuário registrado é qualificado de acordo com as atividades desenvolvidas por ele, os tipos de usuários registrados são desenvolvidos no modelo de usuário e suas atividades são desenvolvidas no modelo de navegação.

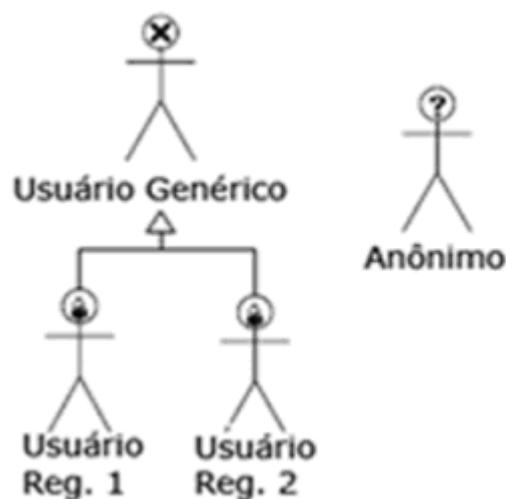


Figura 3 : Modelo de Usuários. Adaptado de Fons (2003) apud Souza (2008)

4.2.3 Modelo de Navegação

O modelo de navegação é dependente do modelo de usuário, já que nele é definido os tipos de usuários. Após a definição dos usuários define-se a semântica de navegação baseado no diagrama de classes que define as atividades do usuário. Um mapa navegacional representa o que o usuário pode “ver” da aplicação e o que pode acessar dependendo do seu tipo. Representado por um grafo o mapa navegacional pode conter em cada nó um contexto navegável ou um subsistema navegável (este é representado como um sub-grafo). Um contexto navegável é representado através de um pacote UML que contém um AIU (*Abstract Information Unit*). A figura 4 mostra um exemplo de um contexto navegacional.

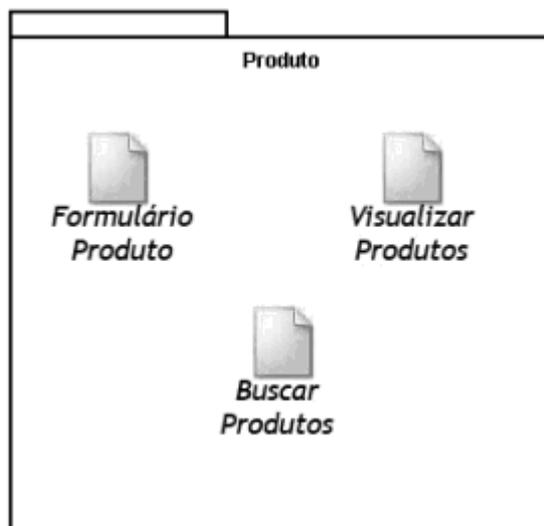


Figura 4 : Contexto Navegacional de produtos de uma aplicação Web

A figura 5 mostra o exemplo de um mapa navegacional, onde o ator tem acesso aos seguintes contextos navegacionais: *Calculabe concepts*, *Entities*, *Metries*, *Tools*, *Attributes* e *Indicators*, nota-se que o acesso a *Indicators* e *Attributes* se dá de forma indireta.

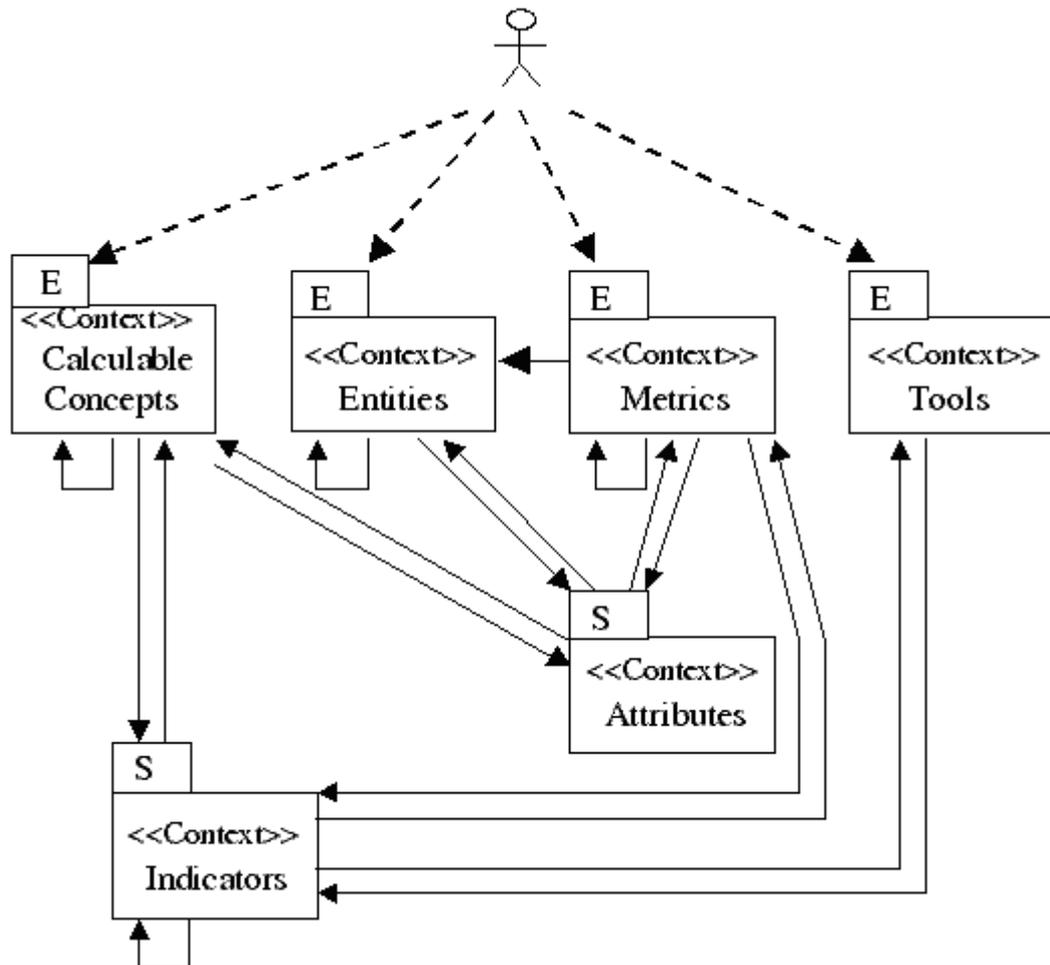


Figura 5 : Exemplo de Modelo de Navegação – Fonte – http://www.w3.org/2001/sw/Europe/reports/dev_workshop_report_8/maria/oows.Modelo.png

4.2.4 Modelo de Apresentação

Este modelo especifica as propriedades que devem cumprir o que está sendo apresentado, para isto é proposto um conjunto de padrões de apresentação baseados nas premissas do modelo navegacional, tal conjunto é chamado de Unidade de Interação (UI) é composto, por exemplo, de como é feita a organização da informação (tabelas, registros).

A Figura 6 ilustra um modelo de apresentação modelado em forma de árvore de ação hierárquica em níveis.

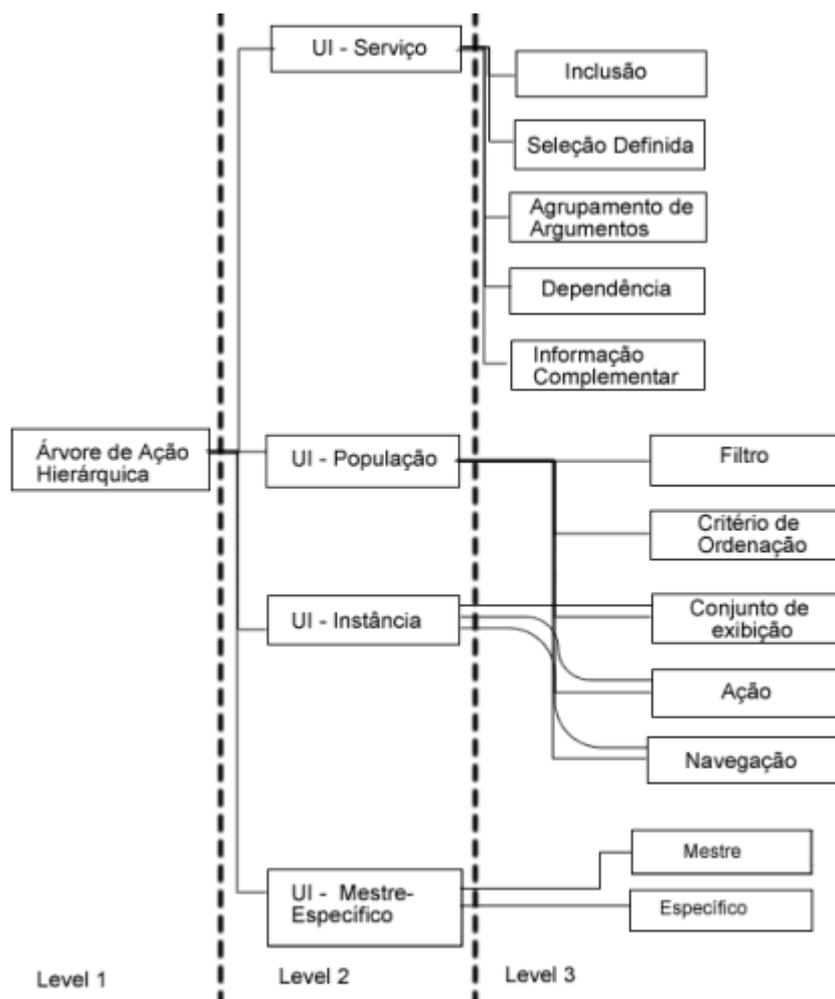


Figura 6 : Árvore de Ação Hierárquica (Souza, 2008)

4.2.5 Modelagem da Arquitetura

Como mostrado antes a modelagem da arquitetura definida no método OOWS possui diversas camadas e especificações que serão definidas a seguir (FONS, 2003 apud Souza, 2008):

Camada de Apresentação: é a camada de interação com o usuário apresentando interfaces gráficas como página de exibição. Essa camada deve ser desenvolvida para os diferentes tipos de usuários.

Camada de Aplicação: define a lógica do negócio permitindo a implementação das estruturas e funcionalidades das classes baseado no modelo conceitual. É dividida em duas subcamadas: (i) interface de negócio

apóia a implementação de aplicações Web, além de atuar como elo entre a camada de persistência e a (ii) lógica do negócio.

Camada de Persistência: armazena e controla os dados omitindo o repositório dos dados nas camadas superiores.

4.3 Método SPL-OOWS

O método de desenvolvimento SPL-OOWS, buscou conceitos da LPS, que é desenvolver aplicações de um domínio através de um conjunto de artefatos com os elementos comuns que podem ser reutilizados e utilizados como um domínio padrão das aplicações web. O foco principal deste método é a criação de uma infra-estrutura padrão e a padronização das etapas para que o desenvolvimento de aplicações web possam ser representadas como LPS.

O SPL-OOWS faz uma expansão das fases do método OOWS incluindo nas etapas conceitos de LPS. As extensões das fases estão principalmente nas etapas de: modelagem conceitual, navegacional e apresentação. A figura 7 apresenta as etapas do método SPL-OOWS podendo ver as expansões propostas por ele, que são: modelo de características ao longo da modelagem conceitual e modelos de infra-estrutura básica na modelagem conceitual e a camada de infra-estrutura básica na arquitetura do desenvolvimento da solução.

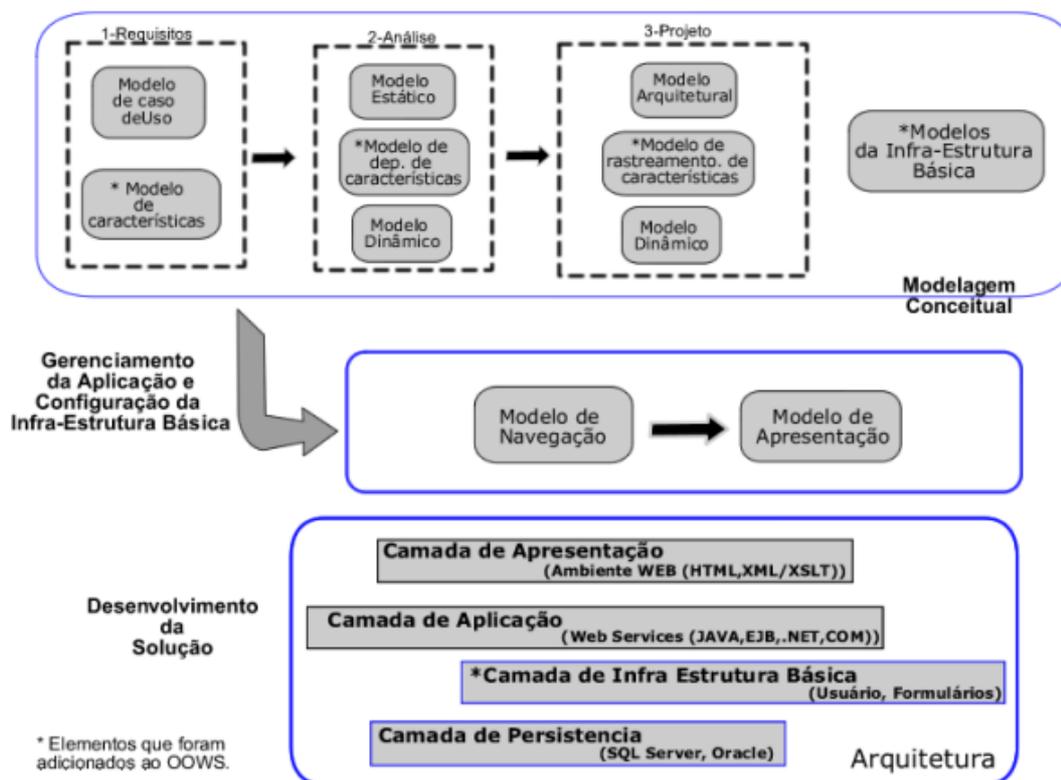


Figura 7 : Modelo SPL-OOWS - Visão Geral (Souza, 2008)

4.3.1 Infra-Estrutura Básica

Um conjunto de elementos comuns para o desenvolvimento de uma aplicação web de qualquer domínio. Tais elementos são configurados e gerenciados com o intuito de serem utilizados para a criação de aplicações específicas. A infra-estrutura básica pode ser estendida para cada domínio, criando assim uma hierarquia de LPS em aplicações web.

O modelo que representa a infra-estrutura básica é mostrado na Figura 8 em forma de diagrama de característica (Czarnecki et al., 2005 apud Souza, 2008). O diagrama é composto de três características principais, o Gerenciamento de Usuários, Gerenciamento de Funcionários e o Gerenciamento de Banco de Dados.

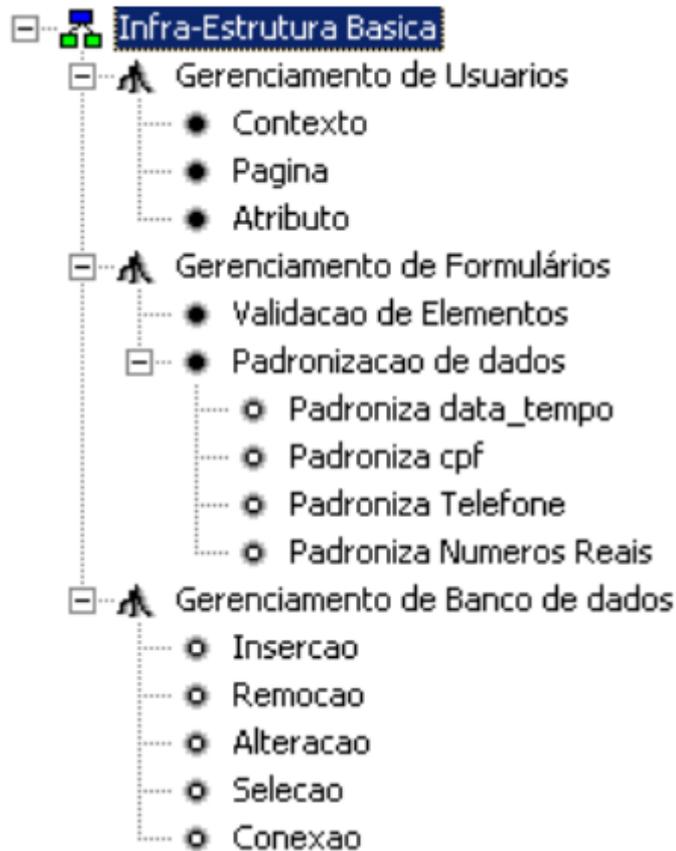


Figura 8 : Infra-estrutura básica (Souza, 2008)

Gerenciamento de Usuário

A SPL-OOWS especifica um modelo de usuário reutilizável para ser configurado para qualquer domínio de aplicação, este modelo é compatível com o OOWS que apresenta aplicações Web compostas de contextos navegacionais fazendo correlação com os usuários que acessam tais contextos. A Figura 9 representa o gerenciamento de Usuário.

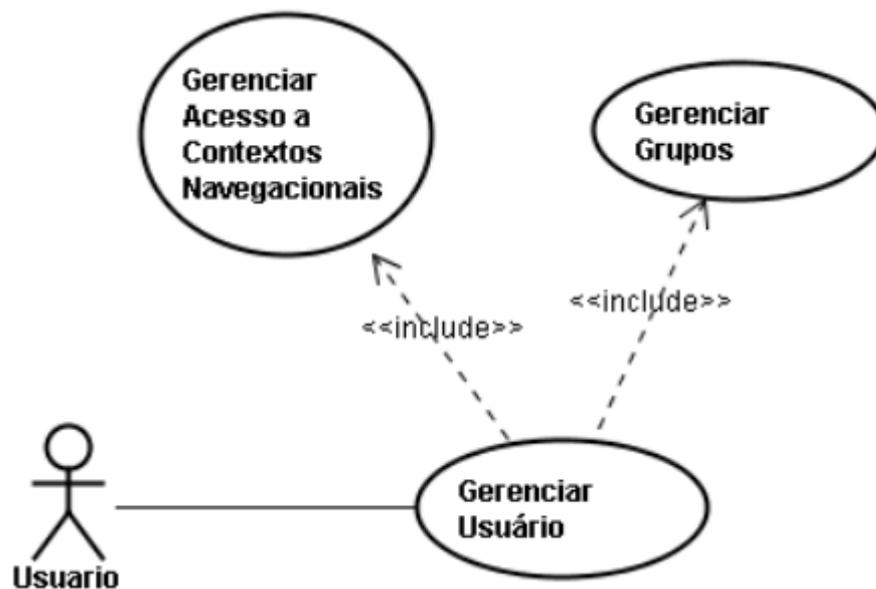


Figura 9: Diagrama de Casos de Uso de Modelo de Usuário (Souza, 2008)

Gerenciamento de Formulários

Nas aplicações Web para se obter a informação sobre usuários são utilizados formulários a serem preenchidos, para um posterior gerenciamento dos mesmos pela lógica da aplicação. O gerenciamento de formulários faz a generalização do fluxo de dados entre os usuários e a lógica da aplicação, já que esta funcionalidade está presente em todas as aplicações web.

Gerenciamento de Banco de Dados

O gerenciamento de banco de dados trata das informações que serão armazenadas em bancos de dados, através de uma interface única, esta interface dará acesso à base de dados independentemente da tecnologia de banco de dados que será implementada.

O gerenciamento fornece as seguintes operações para tratamento das informações: *Insert*(Inserir), *Delete*(Remover), *Update*(Alterar) e *Select*(Visualizar).

4.3.2 Modelagem Conceitual

Para fazer a modelagem conceitual no modelo SPL-OOWS é usada a notação UML estendida para LPS. Para fazer a modelagem foram utilizados: modelo de caso de uso (diagrama), modelo estático (diagrama de classes), modelo dinâmico (diagrama de colaboração e seqüência), diagrama arquitetural e modelo arquitetural (diagrama de componentes).

A modelagem possui três etapas: Requisitos, Análise e Projeto, que serão descritas abaixo.

4.3.2.1 Requisitos

A parte de requisitos se refere ao levantamento dos requisitos funcionais e não-funcionais do sistema, no desenvolvimento de uma LPS, devem ser desenvolvidos os modelos de caso de uso e característica (Gomaa, 2005 apud Souza, 2008).

Modelo de Caso de Uso

A figura 10 mostra um exemplo genérico de um caso de uso.

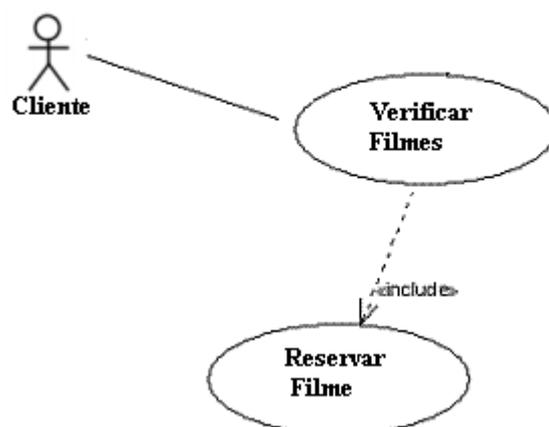


Figura 10 : Exemplo genérico de um caso de uso

Modelo de Característica

O modelo de característica é usado para modelar aplicações de um domínio, especificando as capacidades comuns e diferentes de uma aplicação. É uma técnica de captura e gerenciamento de atributos e variabilidades passando por todo o projeto dentro da LPS. Geralmente os relacionamentos entre pontos de visão e características são representados por atributos.

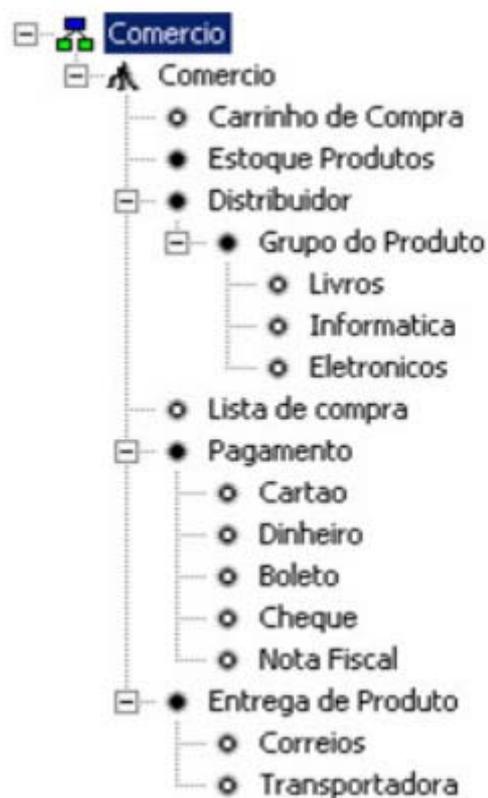


Figura 11 : Modelo de características de comércio eletrônico

4.3.3 Análise

Tem como objetivo manter uma especificação precisa dos requisitos usando linguagem dos desenvolvedores, em que são gerados os modelos estáticos, o modelo dinâmico de estados e o modelo dinâmico interativo.

Modelo Dinâmico de Estados

Este modelo representa o estado dos casos de uso e como é feita a transição desses estados, com base nesse modelo será feito o modelo estático.

Modelo Estático

Este modelo baseado no modelo de estados representa a estrutura estática de uma classe, incluindo os estereótipos de cada classe e os possíveis caminhos alternativos. As classes são divididas em três tipos: *kernel* (são classes fundamentais que estarão presentes em outras aplicações da mesma família de produto), *optional* (são definidas quando seu uso não é fundamental para as outras famílias de produtos) ou *variant* (as classes mudam de acordo com a aplicação a ser desenvolvida).

Modelo Dinâmico Interativo

Este modelo representa a comunicação entre as classes do sistema, tal modelo é representado pelo diagrama de seqüência e deve ser feito um diagrama de seqüência para cada caso de uso especificado na fase de requisitos.

4.3.4 Projeto

Neste momento é feito o mapeamento dos modelos que são desenvolvidos na análise para que os modelos fiquem o mais próximo possível da implementação. Com isso são desenvolvidos os seguintes diagramas: modelo arquitetural, diagrama de seqüência, diagrama de classe (completo) e diagrama de componentes.

Modelo Arquitetural

É dividido em três diagramas: O diagrama de pacotes, diagrama de seqüência e o diagrama de classes.

- Diagrama de Pacotes

Este diagrama é feito baseado na arquitetura de implementação proposta por Fons et al. (2001), porém é adicionado a ele o pacote de infraestrutura básica que contém o gerenciador de usuários, gerenciador de formulários e o gerenciador de banco de dados.

- Diagrama de Seqüência

Define a seqüência em que as operações serão realizadas no sistema, a diferença desse diagrama para o modelo dinâmico interativo é que este possui o nome dos métodos da classe para a realização da tarefa.

- Diagrama de Classe

Mostra como será a estrutura de implementação de solução, o diagrama de classes se difere do modelo estático, pois este há a necessidade de normalizar algumas situações para sanar problemas que podem ser gerados.

- Gerenciamento das Variabilidades

Baseado no modelo de rastreamento de variabilidades é possível identificar quais serão os objetos afetados a partir de uma característica opcional. Assim pode se prever quão boa é a consistência do sistema.

5 Estudo de Caso

Foi desenvolvido um sistema web de locação que gerencia os usuários, funcionários e filmes de uma locadora que provou a eficiência do método SPL-OOWS.

Para isso foi utilizada uma empresa fictícia que desejava implementar uma aplicação web para locação e venda de filmes, com necessidade de agilidade e qualidade, a aplicação foi desenvolvida através do método SPL-OOWS.

Os resultados obtidos serão descritos na seção Conclusão.

5.1 Aplicação do Método SPL-OOWS

Neste capítulo será apresentado o desenvolvimento de uma aplicação para uma locadora de filmes fictícia utilizando o método SPL-OOWS, para o desenvolvimento da aplicação foram utilizadas as seguintes ferramentas: astah community – para a modelagem UML. Para a implementação do programa foram utilizados o servidor Apache, a linguagem de programação PHP e o gerenciador de banco de dados MySQL todos inclusos na ferramenta XAMPP.

5.2 Domínio da Aplicação – Locadora de Filme

O sistema controlará a locação de filmes de uma empresa fictícia com cadastro de usuários, funcionários e filmes, o usuário poderá pesquisar o filme e verificar se o mesmo está disponível ou não. O sistema deverá controlar o acesso dos usuários e controlar a locação de filmes.

O sistema terá 4 (quatro) tipos de usuários: administrador, funcionário, cliente e anônimo.

5.3 Modelagem Conceitual

5.3.1 Requisitos

- Descrição dos requisitos funcionais do sistema
 - Gerenciar funcionários; (alterar e excluir)
 - Cadastrar funcionários;
 - Cadastrar usuários;
 - Gerenciar usuários; (alterar e excluir)
 - Cadastrar filmes;
 - Gerenciar filmes; (alterar e excluir)
 - Reservar filme;
 - Pesquisar filme;
- Descrição dos Requisitos não-funcionais do sistemas
 - Segurança

A Figura 12 representa o modelo de casos de uso feito após o levantamento de requisitos do sistema. Dentro os casos de uso foram definidos como *kernel*: Gerenciar Funcionários, Gerenciar Filmes e Gerenciar Clientes, pois estes são núcleos de qualquer aplicação gerada a partir da LPS. Com isso tem-se os casos de uso Pesquisar Filme e Reservar Filme como *optional*, pois estes não serão usados em qualquer aplicação.

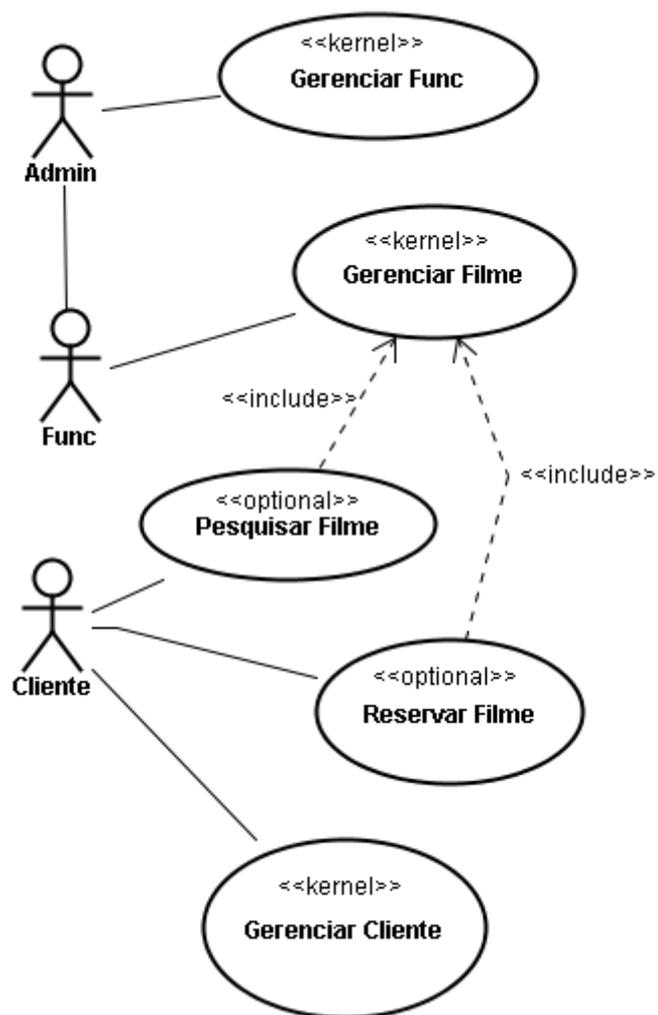


Figura 12: Casos de Uso do Sistema

Após a criação do diagrama de casos de uso será feito o modelo de características do sistema, a Figura 13 representa tal modelo.

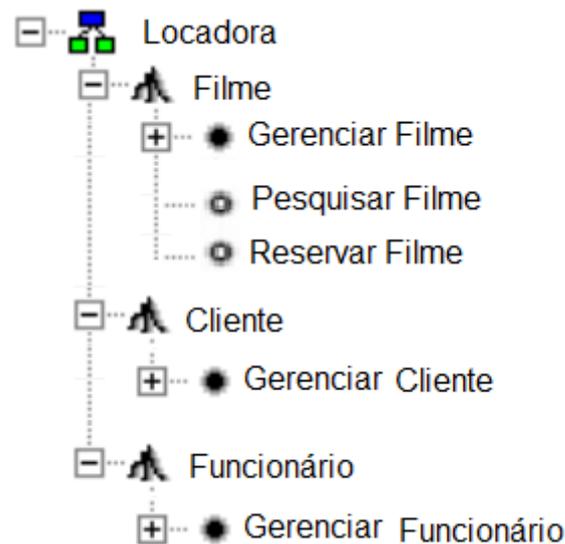


Figura 13: Modelo de Características do Sistema

5.3.2 Análise

Esta subseção apresentará os resultados obtidos através da etapa de análise.

Modelo Dinâmico de Estado

O modelo de estado mostra os estados que serão realizados durante a reserva de um filme. A Figura 14 representa o modelo de estado do caso de uso Gerenciar Funcionário, somente o administrador poderá utilizar esta função do sistema, após logado o administrador poderá alterar, cadastrar ou excluir um funcionário.

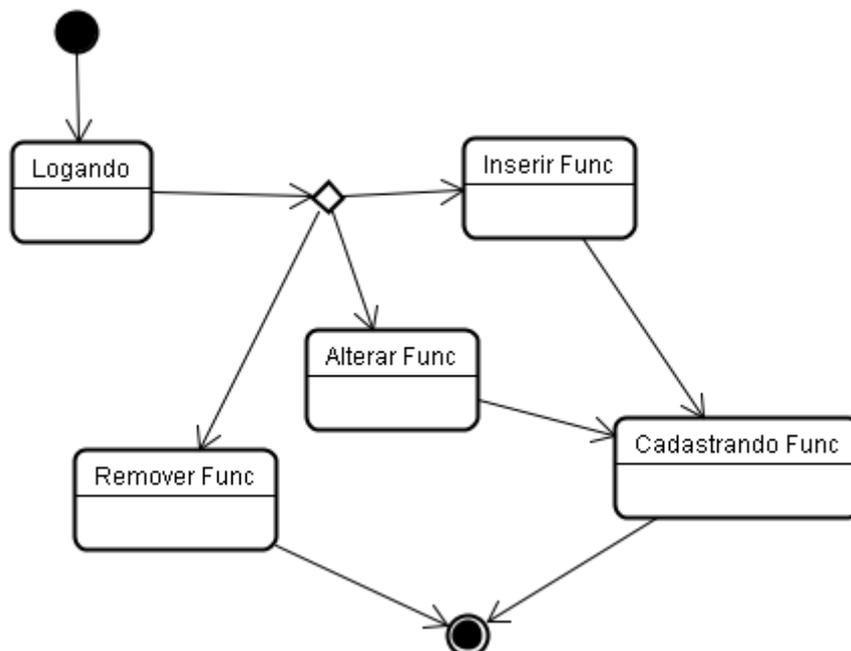


Figura 14: Modelo de Estado do Caso de Uso Gerenciar Funcionário

A Figura 15 representa o modelo de estado do caso de Gerenciar Filme, somente o funcionário e o administrador poderão utilizar esta função, após logado eles poderão alterar, inserir ou remover um filme.

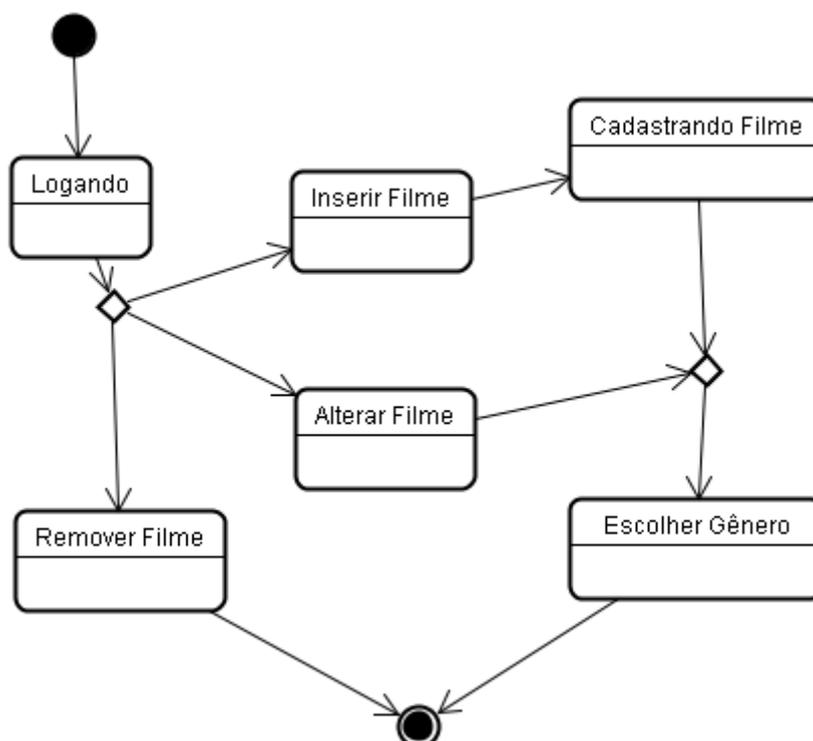


Figura 15: Modelo de Estado do Caso de Uso Gerenciar Filme

A Figura 16 apresenta o modelo de estado do caso de uso Gerenciar Usuário, todos terão acesso a esta seção, o usuário irá clicar em Cadastrar, após o clique ele se cadastrará no sistema e após seu cadastro efetuando o login ele poderá alterar ou excluir seu cadastro.

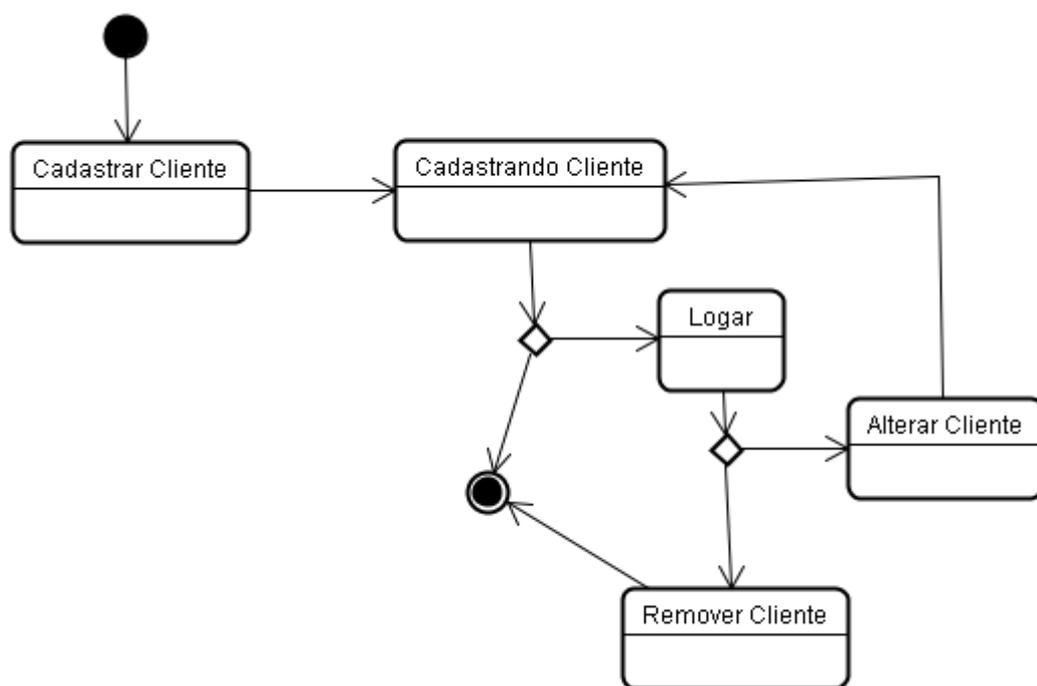


Figura 16: Modelo de Estado do Caso de Uso Gerenciar Cliente

A Figura 17 mostra o modelo de estado gerado para o caso de uso Pesquisar Filme, todos terão acesso a esta seção até mesmo usuários anônimos, o usuário irá pesquisar o filme, podendo ou não fazer o login no sistema para fazer a reserva do mesmo.

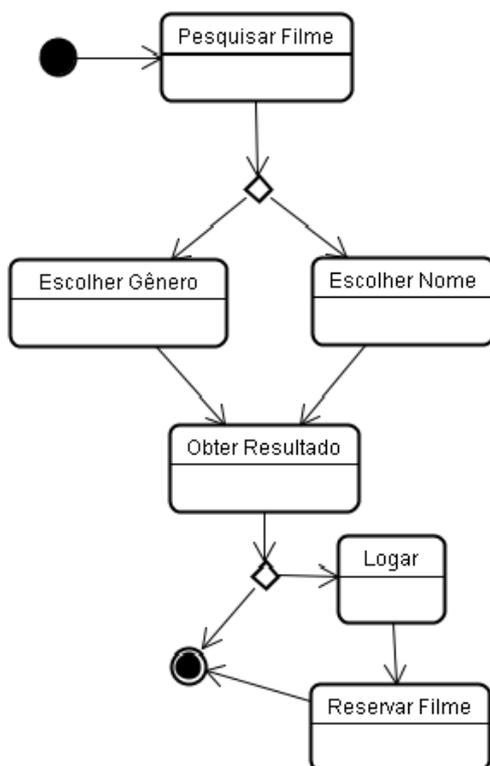


Figura 17: Modelo de Estado do Caso de Uso Pesquisar Filme

A Figura 18 mostra o modelo de estado do caso de uso Reservar Filme, somente usuários logados terão acesso a esta seção, o usuário deverá logar para poder fazer a reserva do filme escolhido.

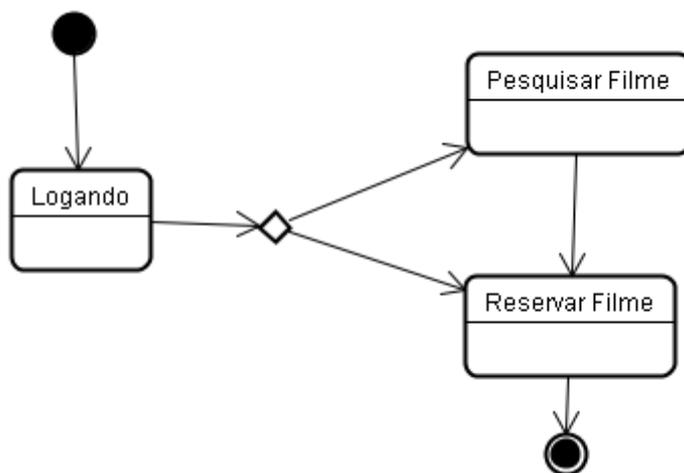


Figura 18: Modelo de Estado do Caso de Uso Reservar Filme

Modelo Estático

O modelo estático mostra as ações vinculadas às classes do sistema, mostrando os pontos de variação (*variant*), opcionais (*optinal*) e de núcleo (*kernel*). A Figura 19 mostra o modelo estático.

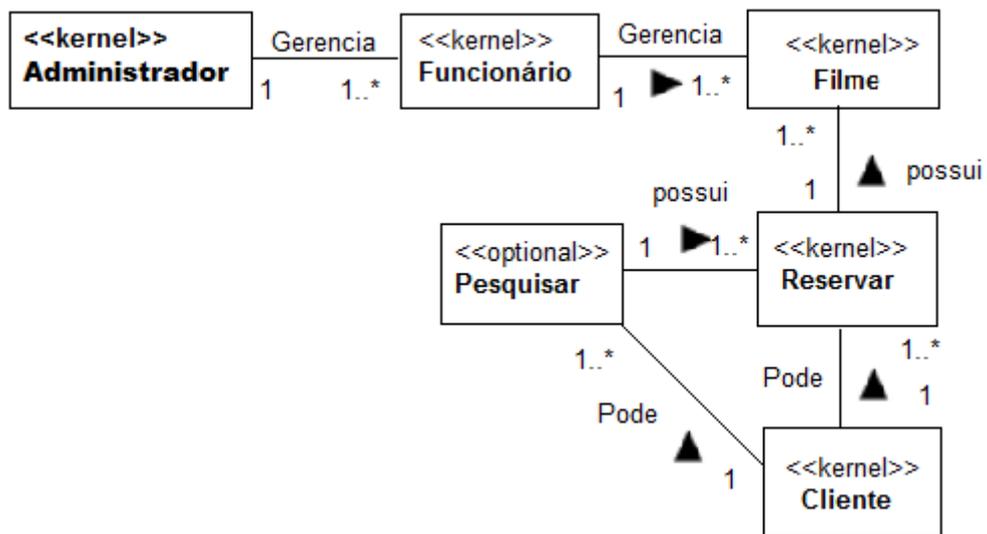


Figura 19: Modelo Estático do Sistema

Modelo Dinâmico Interativo

O modelo dinâmico interativo é representado pelo diagrama de seqüência, a Figura 20 demonstra tal diagrama representando o caso de uso Gerenciar Cliente.

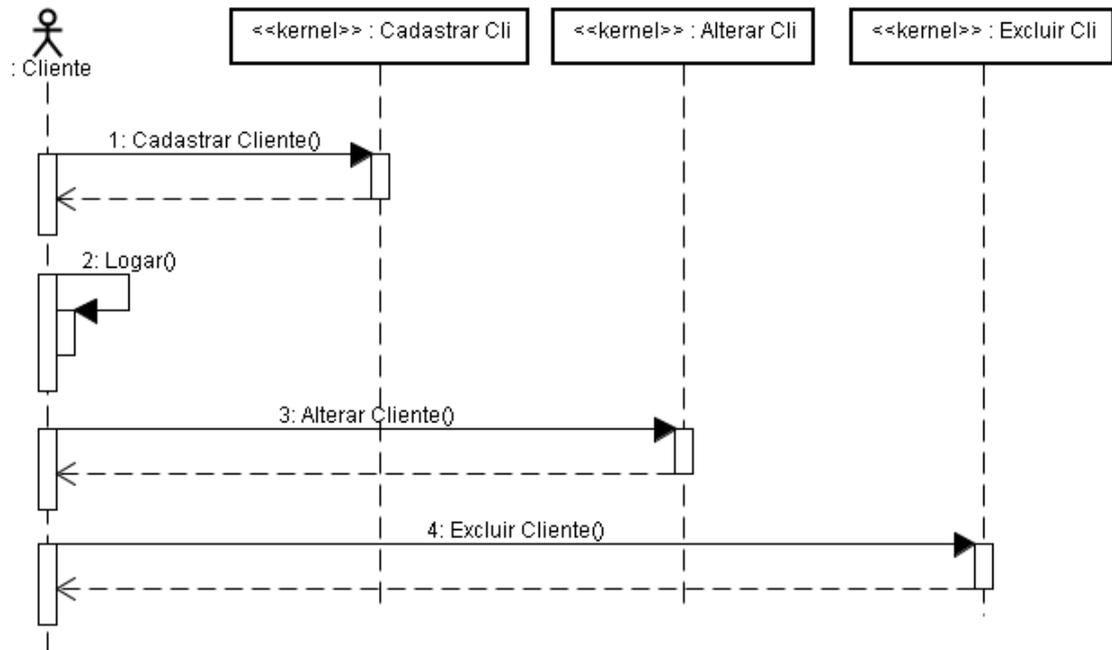


Figura 20: Modelo Dinâmico Interativo do Gerenciar Cliente

A Figura 21 mostra o diagrama do caso de uso Gerenciar Filme.

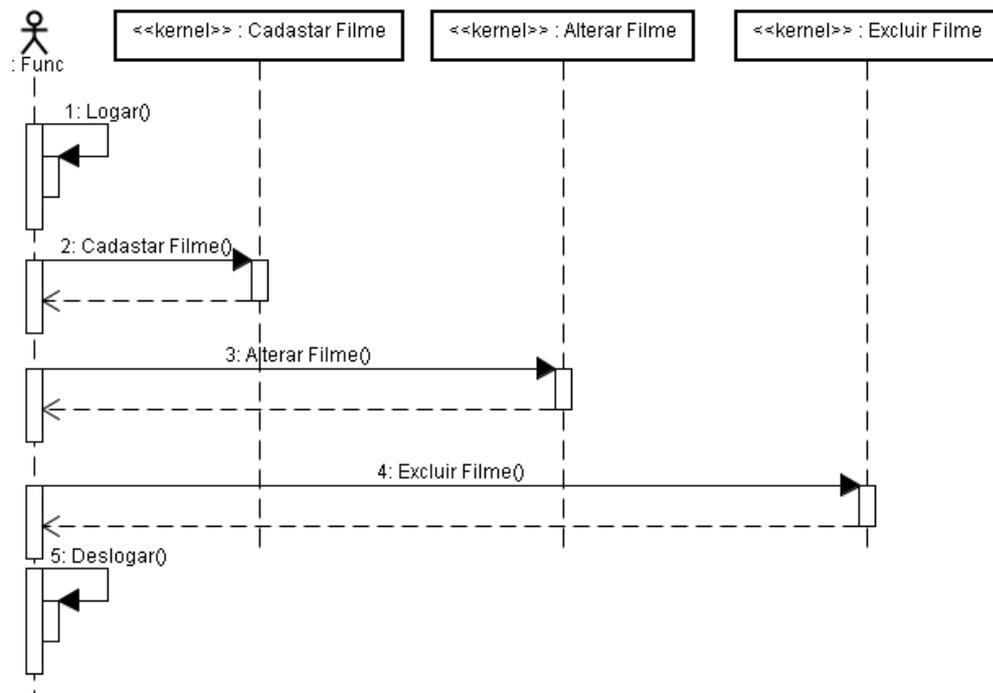


Figura 21: Modelo Dinâmico Interativo do Gerenciar Filme

A Figura 22 mostra o diagrama do caso de uso Gerenciar Funcionário

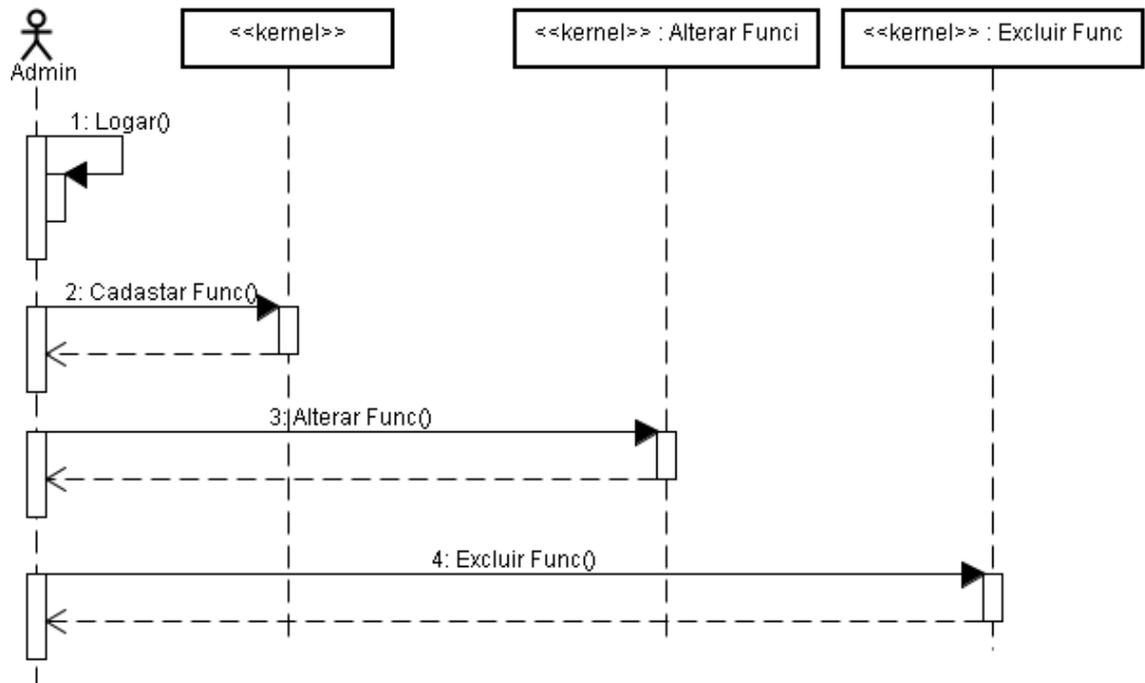


Figura 22: Modelo Dinâmico Interativo do Gerenciar Funcionário

A Figura 23 mostra o modelo dinâmico do caso de uso Pesquisar Filme.

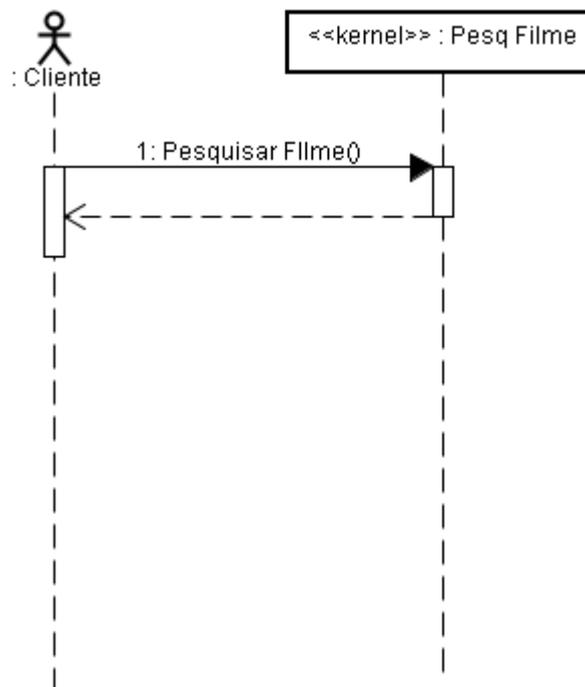


Figura 23: Modelo Dinâmico Interativo do Pesquisar Filme

A Figura 24 mostra o modelo dinâmico interativo do caso de uso Reservar Filme.

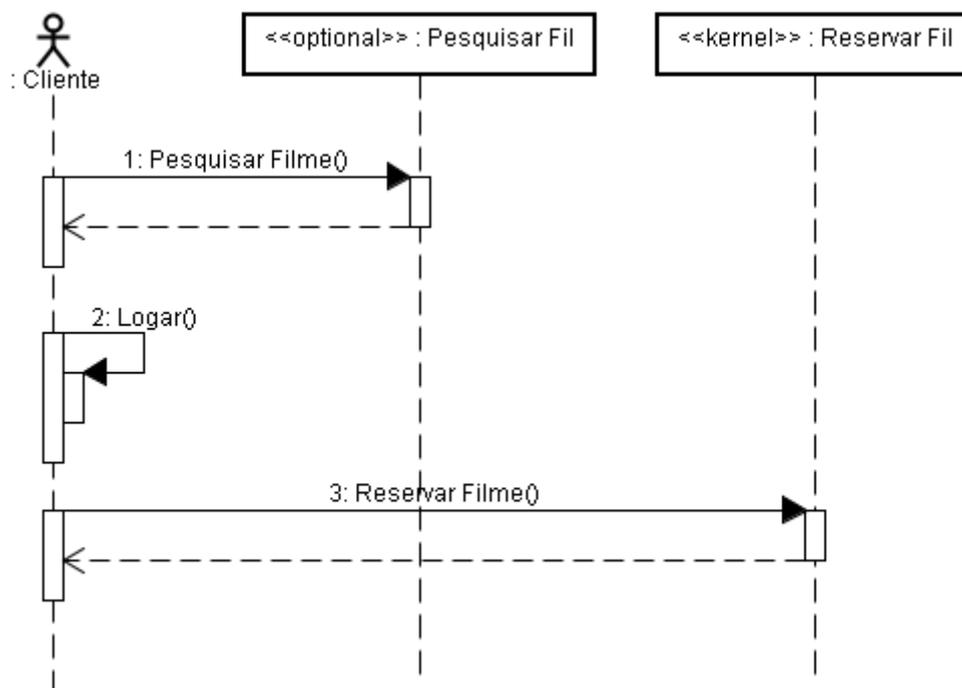


Figura 24: Modelo Dinâmico Interativo do Reservar Filme

Modelo de Rastreamento de Variabilidades

O modelo permite ver quais serão os objetos afetados a partir de uma característica opcional. A Tabela 1 demonstra esse modelo.

Tabela 1: Modelo de Rastreamento de Variabilidades

FEATURES/ CASOS DE USO	GERENCIAR FILME	GERENCIAR CLIENTE	GERENCIAR FUNCIONÁRI	PESQUISAR FILME	RESERVAR FILME
GERENCIAR FILME	•			•	•
GERENCIAR CLIENTE		•			
GERENCIAR FUNCIONÁRIO			•		

5.3.3 Projeto

Modelo Arquitetural

O modelo arquitetural do programa segue a lógica explicada na seção 4.3, este modelo é representado pelos seguintes diagramas: diagrama de pacotes, diagrama de seqüência e o diagrama de classes. A Figura 25 representa o diagrama de seqüência do caso de uso Gerenciar Cliente.

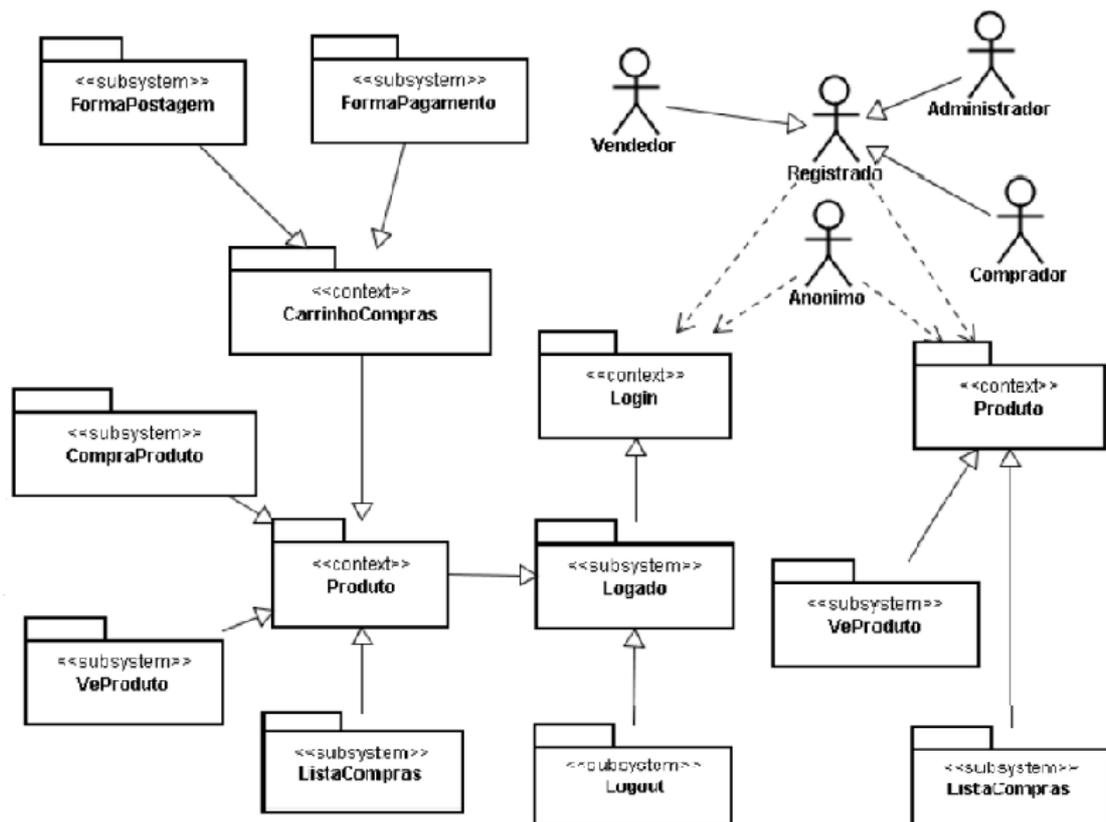


Figura 25: Modelo Geral do Comércio Eletrônico (Souza, 2008)

A Figura 26 representa o diagrama do caso de uso Gerenciar Filme.

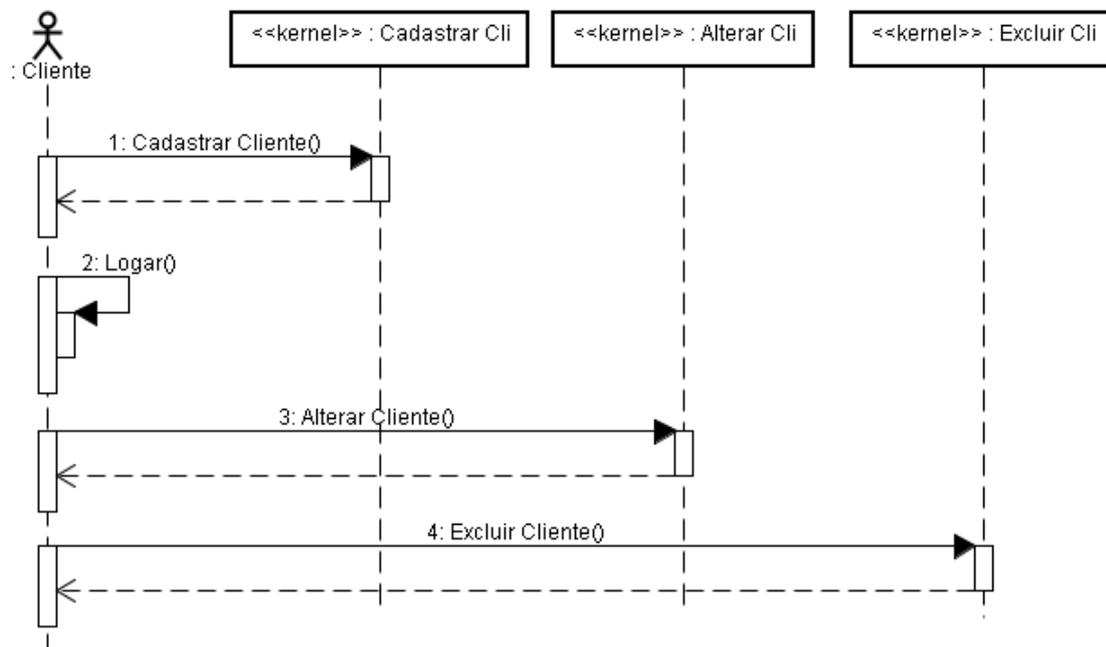


Figura 26: Diagrama de Seqüência de Gerenciar Cliente

A Figura 27 representa o diagrama do caso de uso Gerenciar Filme.

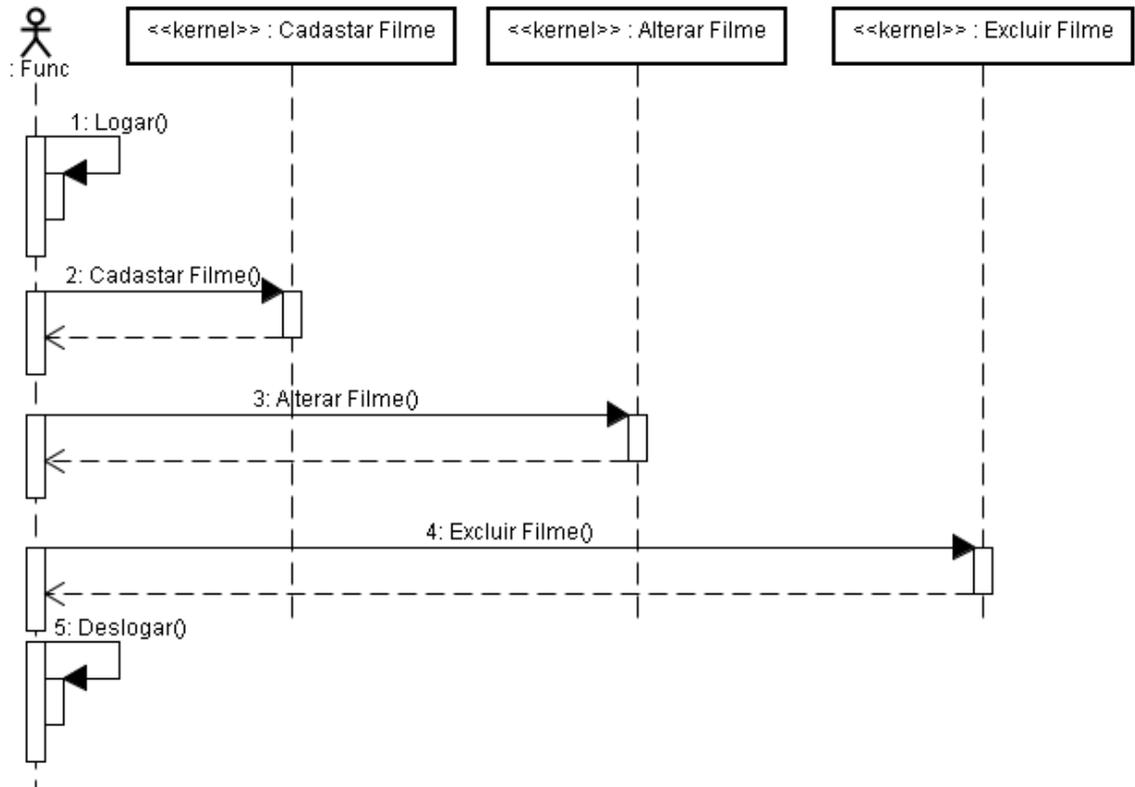


Figura 27: Diagrama de Seqüência do Gerenciar Filme

A Figura 28 representa o diagrama de seqüência do Gerenciar Funcionário.

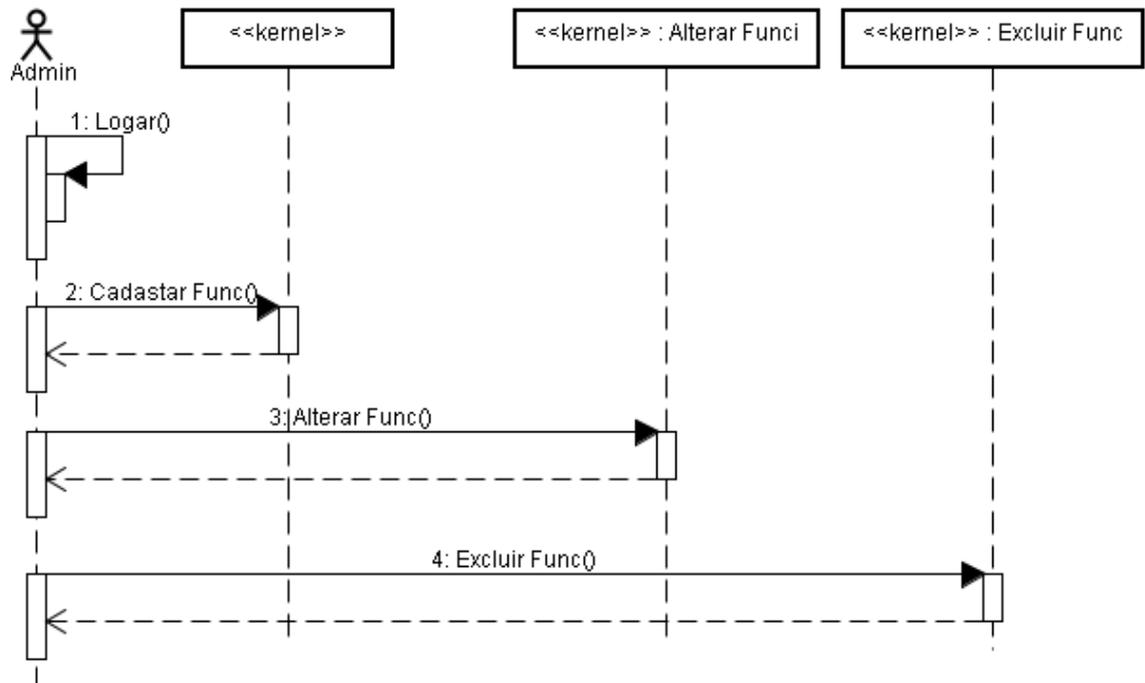


Figura 28: Diagrama de Seqüência do Gerenciar Funcionário

A Figura 29 representa o diagrama de seqüência de Pesquisar Filme.

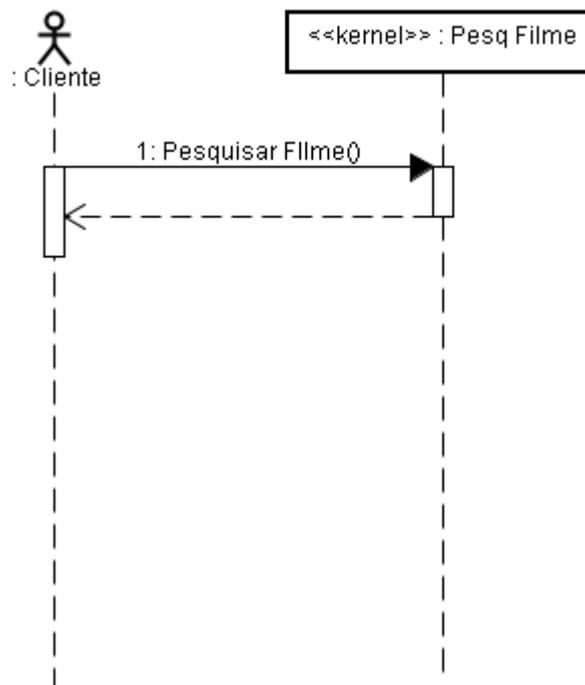


Figura 29: Diagrama de Seqüência de Pesquisar Filme

A Figura 30 representa o diagrama de seqüência de Reservar Filme

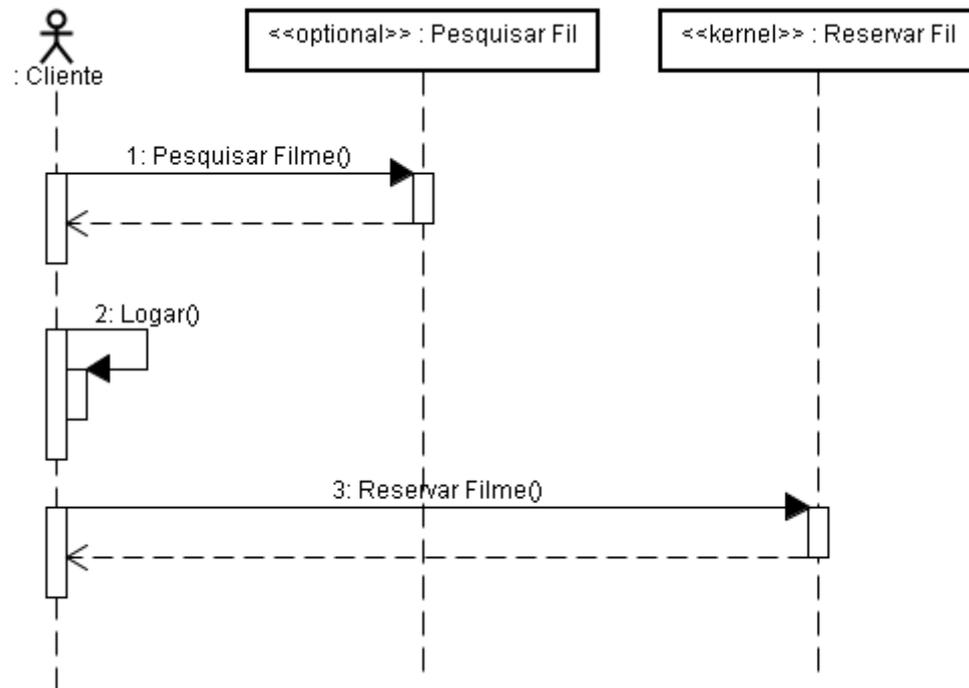


Figura 30: Diagrama de Seqüência de Reservar Filme

A Figura 31 representa o diagrama de classes do sistema juntamente com seus métodos, atributos e a estrutura lógica da aplicação.

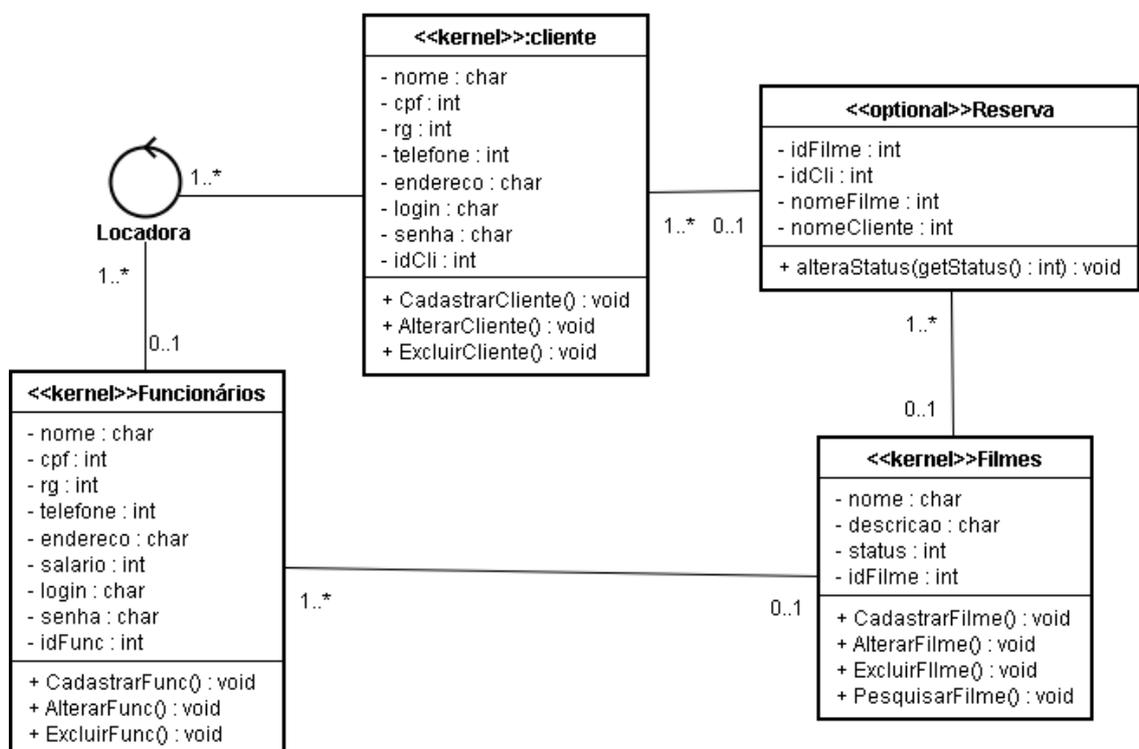


Figura 31: Diagrama de Classes da Locadora

A figura 32 representa como o sistema ficou organizado em pastas.

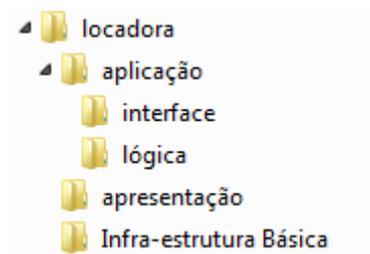


Figura 32: Estrutura de Implementação do Sistema

5.4 Configuração da Aplicação

Nesta seção o primeiro modelo a ser configurado é o modelo de navegação dos usuários do sistema. A Figura 33 mostra o modelo de navegação do Administrador e dos Funcionários do sistema, somente o administrador terá permissão para acessar o contexto Gerenciar Funcionários.

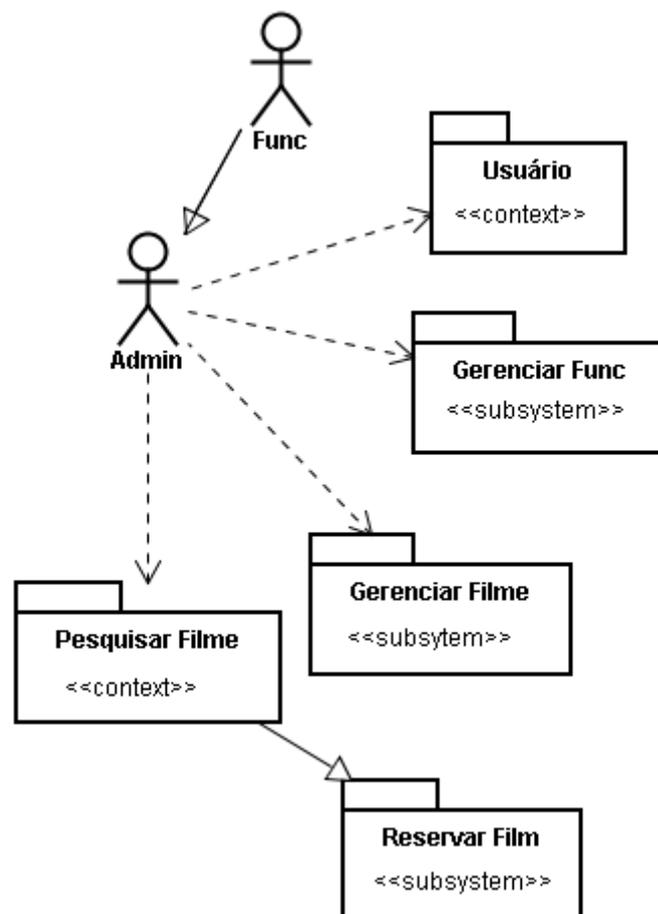


Figura 33: Modelo de Navegação do Administrador e Funcionário

A Figura 34 mostra o modelo do usuário Cliente, este terá acesso aos contextos Gerenciar Cliente, Reservar Filme e Pesquisar Filme.

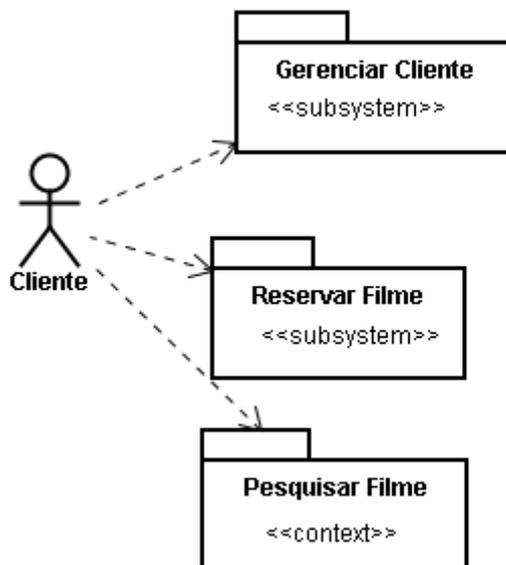


Figura 34: Modelo de Navegação do Cliente

A Figura 35 mostra o modelo de navegação do usuário Anônimo, este só terá acesso ao contexto Pesquisar Filme.

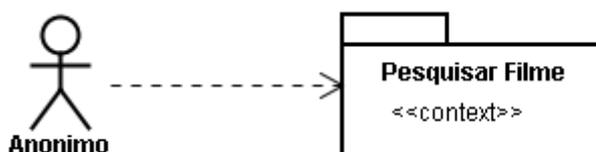


Figura 35: Modelo de Navegação Pesquisar Filme

Após os estudos sobre os modelos de navegação é feito o mapeamento dos contextos navegacionais e as características da aplicação para a identificação de quais modelos serão afetados a partir de uma configuração no modelo de características.

Tabela 2: Mapeamento dos contextos navegacionais e as características da aplicação

Características	Contextos Navegacionais
Gerenciar Filme	Gerenciar Filme
Gerenciar Funcionário	Gerenciar Funcionário
Gerenciar Usuário	Gerenciar Usuário

Pesquisar Filme	Pesquisar Filme
-----------------	-----------------

A partir dos modelos definidos será feito uma árvore de ação hierárquica para cada um que definirão com certeza quais serão os elementos da apresentação da aplicação.

A Tabela 3 representa a árvore de ação hierárquica para o contexto Gerenciar Funcionário.

Tabela 3: Árvore de Ação Hierárquica para o contexto Gerenciar Funcionário

Nível 2	Nível 3	Descrição
Serviços UI	Validação de Campos	Gerenciador de Usuários
	Lista de Seleção	Listar Funcionários, Buscar Funcionário
Instância UI	Conjunto de Exibição	Nome,CPF,código,endereço, RG, telefone
	Ações	Cadastrar Funcionário, Alterar Funcionário, Excluir Funcionário, Listar Funcionário
	Navegação	Gerenciar Filme, Pesquisar Filme, Gerenciar Usuário
População UI	Filtro	Funcionário por CPF, Funcionário por código
	Conjunto de Exibição	Query Lista Funcionário
	Ações	Cadastrar Funcionário, Alterar Funcionário, Excluir Funcionário, Listar Funcionário

	Navegação	Gerenciar Filme, Pesquisar Filme, Gerenciar Usuário
Mestre- Específico UI	Mestre	CPF, código
	Específico	RG, nome

A Tabela 4 demonstra a árvore de ação hierárquica para o contexto Gerenciar Cliente.

Tabela 4: Árvore de Ação Hierárquica para o contexto Gerenciar Cliente

Nível 2	Nível 3	Descrição
Serviços UI	Validação de Campos	Gerenciador de Usuários
	Lista de Seleção	Listar Cliente, Buscar Cliente
Instância UI	Conjunto de Exibição	Nome, CPF, código, endereço, RG, telefone, login
	Ações	Cadastrar Cliente, Alterar Cliente, Excluir Cliente, Listar Cliente
	Navegação	Pesquisar Filme, Gerenciar Usuário, Reservar Filme
População UI	Filtro	Cliente por login, Funcionário por código
	Conjunto de Exibição	Query Lista Funcionário
	Ações	Cadastrar Cliente, Alterar Cliente, Excluir Cliente, Listar Cliente

	Navegação	Pesquisar Filme, Gerenciar Usuário, Reservar Filme
Mestre- Específico UI	Mestre	CPF, código
	Específico	RG, nome

A Tabela 5 apresenta a árvore de ação hierárquica de Gerenciar Filme.

Tabela 5: Árvore de Ação Hierárquica para o contexto Gerenciar Filme

Nível 2	Nível 3	Descrição
Serviços UI	Validação de Campos	Gerenciador de Formulários
	Lista de Seleção	Listar Filme, Buscar Filme
Instância UI	Conjunto de Exibição	Nome, Gênero, Status, Código
	Ações	Cadastrar Filme, Alterar Filme, Excluir Filme, Listar Filme
	Navegação	Pesquisar Filme, Reservar Filme
População UI	Filtro	Filme por gênero, filme por status
	Conjunto de Exibição	Query Lista Filmes
	Ações	Cadastrar Filme, Alterar Filme, Excluir Filme, Listar Filme
	Navegação	Pesquisar Filme, Reservar Filme

UI	Mestre-Específico	Mestre	Código
		Específico	Status

A Tabela 6 demonstra a árvore de ação hierárquica do contexto Reservar Filme.

Tabela 6: Árvore de Ação Hierárquica para o contexto Reservar Filme

Nível 2	Nível 3	Descrição	
Serviços UI	Validação de Campos	Gerenciador de Formulários	
	Lista de Seleção	Reservar Filme	
Instância UI	Conjunto de Exibição	Nome, Gênero, Status, Código	
	Ações	Reservar Filme	
	Navegação	Pesquisar Filme, Reservar Filme	
População UI	Filtro	Filme por gênero, filme por status	
	Conjunto de Exibição	Query Lista Filmes	
	Ações	Reservar Filme	
	Navegação	Pesquisar Filme, Reservar Filme	
UI	Mestre-Específico	Mestre	Código
		Específico	Status

A Tabela 7 apresenta a árvore de ação hierárquica do contexto Pesquisar Filme.

Tabela 7: Árvore de Ação Hierárquica para o contexto Pesquisar Filme

Nível 2	Nível 3	Descrição
Serviços UI	Validação de Campos	Gerenciador de Formulários
	Lista de Seleção	Pesquisar Filme
Instância UI	Conjunto de Exibição	Nome, Gênero, Status, Código
	Ações	Pesquisar Filme
	Navegação	Pesquisar Filme, Reservar Filme
População UI	Filtro	Filme por gênero, filme por status
	Conjunto de Exibição	Query Lista Filmes
	Ações	Pesquisar Filme
	Navegação	Pesquisar Filme, Reservar Filme
Mestre-Específico UI	Mestre	Código
	Específico	Status

Considerações finais, após a utilização do método foi possível perceber que o método a princípio exige um pouco mais de tempo para o entendimento do mesmo e também que é necessário um pouco de experiência do usuário para um melhor aproveitamento do método.

6 Conclusão

Ao iniciar o desenvolvimento do sistema utilizando o método descrito neste trabalho o mesmo aparenta ser de difícil utilização, porém ao se definir quais são os tipos de classes (vide seção 4.3.3) o método começa a ficar de fácil compreensão. Após a elaboração dos diagramas e tabela que são exigidos pelo método a implementação se torna extremamente fácil, pois a comunicação entre as classes e o que cada classe deve conter já estão definidas.

Algumas dificuldades foram percebidas durante a utilização do SPL-OOWS, tais como, a elaboração do modelo estático do sistema, compreensão de como funciona a infra-estrutura básica do método e elaboração do modelo de características do sistema.

Uma das vantagens de se utilizar o método são os modelos navegacionais, que explicam precisamente quais serão os contextos que cada usuário poderá acessar. Além das árvores de ação hierárquica que definem quais serão os elementos de apresentação do sistema.

Porém os processos de descrições, diagramações e desenvolvimento do sistema analisado como um todo sob a perspectiva do método não faz o método ser algo de difícil utilização, mas, no entanto, o usuário do mesmo necessita de um pouco de conhecimento em desenvolvimento de aplicações desse tipo.

Com isso, pode-se perceber que ao se utilizar o método para o desenvolvimento de um sistema de um domínio de aplicação pode tomar o tempo que os outros métodos de desenvolvimento utilizam, mas, para o possível desenvolvimento de outro sistema da mesma família de produtos, o tempo pode ser encurtado, pois todos os diagramas, descrições e contextos já foram definidos no desenvolvimento do domínio da aplicação.

Com o domínio da aplicação criado para ser utilizado como um Gerenciador de Locadora, outros sistemas da mesma linha de produtos podem ser desenvolvidos através desse mesmo domínio, como, por exemplo, um sistema para locação de livros em uma biblioteca, onde o funcionário e o administrador gerenciam os livros, o próprio usuário se cadastra, pesquisa e reserva o livro que deseja, ou então, um sistema de gerenciamento de locação de quartos em uma pousada.

Após a análise do método chega-se a conclusão que o método supera outros métodos de desenvolvimento nos quesitos tempo e qualidade dos sistemas da mesma família de produtos do domínio da aplicação, pois os erros cometidos no desenvolvimento do domínio já estarão corrigidos no desenvolvimento dos demais sistemas, além das melhorias que podem ser implementadas nos demais sistemas da família.

Referências

DURSCKI, Roberto C. et al. **Linhas de Produto de Software: riscos e vantagens de sua implantação**, 2004. Disponível em: <http://www.dimap.ufrn.br/~andre/gti-16/linha_de_producao.pdf>. Acesso em: 24 de maio de 2010.

LOBO, Ana Elisa de Campos; RUBIRA, Cecília Mary Fischer. **Um Estudo para Implantação de Linha de Produto de Software baseada em Componentes**, 2009. Disponível em: <<http://www.ic.unicamp.br/~reltech/2009/09-17.pdf>>. Acesso em: 15 de maio de 2010.

SBBD/2006 – SBES/2006. **Linhas de Produto de Software: Reuso que faz Sentido para Negócios**. Disponível em: <<http://www.inf.ufsc.br/sbes2006/linda.htm> >. Acesso em: 17 de maio de 2010.

Souza, Bruno M. N. **Uma extensão do método OOWS para linha de produto de software**, 2008.

TEIXEIRA, Leopoldo Motta. **Ligo: Uma linha de produtos de software para gerenciamento de igrejas cristãs**, 2007.

VALVERDE, Francisco; VALDERAS, Pedro; Fons, Joan. **OOWS Suite: Un Entorno de desarrollo para Aplicaciones Web basado en MDA**, 2007. Disponível em: <http://kuainasi.ciens.ucv.ve/ideas07/documentos/articulos_ideas/Articulo34.pdf>. Acesso em 19 de maio de 2010.

ZAUPA, Fábio et al. **Um processo de desenvolvimento de Aplicações Web baseado em serviços**, 2007.