

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO**

**SISTEMA DE INDICAÇÃO DE PRODUTOS ATRAVÉS DE**  
**RETENÇÃO DE CONHECIMENTO**

**JONATHAN FRANCESCO STEILEIN**

**BLUMENAU**  
**2006**

**2006/1-13**

**JONATHAN FRANCESCO STEILEIN**

**SISTEMA DE INDICAÇÃO DE PRODUTOS ATRAVÉS DE  
RETENÇÃO DE CONHECIMENTO**

Trabalho de Conclusão de Curso submetido à  
Universidade Regional de Blumenau para a  
obtenção dos créditos na disciplina Trabalho  
de Conclusão de Curso II do curso de Sistemas  
de Informação — Bacharelado.

Prof. Oscar Dalfovo , Dr. – Orientador

**BLUMENAU  
2006**

**2006/1-13**

# **SISTEMA DE INDICACAO DE PRODUTOS ATRAVÉS DE RETENÇÃO DE CONHECIMENTO**

Por

**JONATHAN FRANCESCO STEILEIN**

Trabalho aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II, pela banca examinadora formada por:

Presidente:

---

Prof. Oscar Dalfovo, Dr. – Orientador, FURB

Membro:

---

Prof. Mauro Marcelo Mattos, Dr. – FURB

Membro:

---

Prof. Jomi Fred Hubner, Dr. – FURB

Blumenau, 30 de Agosto de 2006

Dedico este trabalho a todos os amigos, especialmente aqueles que me ajudaram diretamente na realização deste. Aos professores que me apoiaram nesta caminhada.

## **AGRADECIMENTOS**

À Deus, pelo seu imenso amor e graça, obrigado por tudo.

À minha família, que de seu modo, sempre depositou suas fichas em mim.

À minha namorada Marieli Mota da Rosa que sempre me apoiou e cobrou meus deveres.

Aos meus amigos, simplesmente os melhores que uma pessoa poderia ter, que compreenderam minha ausência.

Ao meu orientador, Oscar Dalfovo, por ter acreditado na conclusão deste trabalho e durante todo o curso esteve presente demonstrando grande apoio.

## RESUMO

Neste trabalho é tratada uma abordagem entre CRM e os Agentes de *interface*, onde ambos colaboram para tentar passar a imagem de exclusividade ao cliente através da recomendação de títulos direcionados exclusivamente a ele. Depois de concluída a implementação do sistema, o mesmo foi implantado em um comércio da região de Blumenau, onde seletos clientes o utilizaram e comprovaram sua eficiência.

Palavras-chave: Customer relationship management. Agentes de *interface*.

## **ABSTRACT**

This work describe a video rental store software based on CRM (Customer relationship management) and the interface Agents concepts in order to provide a kind of personalization of the movies suggestions to store's clients.

Key-words: Customer relationship management. Interface agents.

## LISTA DE ILUSTRAÇÕES

Figura 1 : Definição de agentes .....	15
Quadro 1 – Requisitos funcionais.....	21
Quadro 2 –Requisitos não funcionais.....	22
Figura 2 – Diagrama de caso de uso do usuário .....	23
Figura 3 –Diagrama de classe.....	24
Quadro 3 – Descrição das classes utilizadas na aplicação.....	25
Figura 4 – Modelo físico de dados .....	26
Figura 5 – Modelo físico de dados (continuação) .....	27
Figura 6 – diagrama de atividades.....	28
Figura 7 – métodos da classe Conecta.....	30
Figura 8 – Classe Gênero .....	31
Figura 9 – métodos da classe AIC .....	32
Figura 10 – método set_novos_parametros(\$sql).....	34
Figura 11 – Página inicial da aplicação .....	34
Figura 12– Menu principal .....	35
Figura 13 – Consulta por títulos dos filmes.....	35
Figura 14 – Futuros lançamentos.....	36
Figura 15 – Os mais locados .....	37
Figura 16 – Menu de navegação do cliente .....	37
Figura 17– Cartão com senha do cliente .....	38
Figura 18 – Tela para definição das preferências primárias do cliente .....	39
Figura 19 – AII recomenda títulos ao cliente logado .....	40
Figura 20 – Cadastro do cliente do programa local.....	41

## SUMÁRIO

1.1 OBJETIVOS DO TRABALHO .....	11
1.2 ESTRUTURA DO TRABALHO .....	11
2.1 INFORMAÇÃO E CONHECIMENTO .....	13
2.2 AGENTES DE SOFTWARE .....	14
2.2.1 AGENTES DE INTERFACE .....	15
2.3 CUSTOMER RELANTIONSHIP MANAGEMENT (CRM) .....	16
2.3.1 INTRODUÇÃO .....	17
2.3.2 CONCEITO .....	17
2.3.3 APLICAÇÃO DO CRM .....	18
2.3.4 RETORNO DO USO DO CRM .....	18
3.1 ESPECIFICAÇÃO .....	20
3.2 PROJETO .....	20
3.2.1 REQUISITOS .....	21
3.2.2 CASOS DE USO .....	22
3.2.3 DIAGRAMA DE CLASSES .....	23
3.2.4 DIAGRAMA FÍSICO DE DADOS .....	25
3.2.5 DIAGRAMA DE ATIVIDADES .....	28
3.3 TECNOLOGIAS E FERRAMENTAS UTILIZADAS.....	29
3.3.1 PHP .....	29
3.4 IMPLEMENTAÇÃO DA APLICAÇÃO.....	29
3.4.1 ATUAÇÃO DO AGENTE AII.....	33
3.5 APRESENTAÇÃO DA APLICAÇÃO.....	34
3.6 RESULTADO .....	41
4.2 EXTENSÕES .....	43

## 1 INTRODUÇÃO

No chamado atual paradigma da informação, prover e reter a informação, que já é considerado um bem da empresa, tem se tornado cada vez mais difícil. “Neste sentido, as organizações têm investido intensamente em projetos que objetivam a criação, o controle e a disponibilização adequada da informação no ambiente de trabalho” (RIOS, 2005, p.2).

Segundo Cardoso e Gonçalves Filho (2001), a informação nas organizações é hoje um poderoso recurso, permitindo seu perfeito alinhamento estratégico por meio de constantes fluxos bidirecionais entre a empresa e o macro-ambiente, criando condições para que esta viabilize seus objetivos e cumpra sua missão corporativa.

Em contra partida, segundo Weitzen (1991), é necessário haver discrepância entre as informações essenciais e não essenciais para o negócio, onde criou-se atualmente uma sobrecarga de informações, e mesmo as informações úteis devem ser organizadas e apresentadas de forma sistemática para tal discernimento.

Para Carvalho e Tavares (2001) a informação essencial é aquela que contribui diretamente para com o negócio da organização e circula, dentro dela, de forma limpa, sistematizada e racional.

Ainda segundo Carvalho e Tavares (2001), assim como uma pessoa pode ter acesso a inúmeras informações, mas se não souber colocá-las em um contexto, interpretá-las e traduzi-las em uma nova realidade, não estará gerando conhecimento, sendo assim necessário tal armazenamento e disseminação de informação em gerando conhecimento.

Felipini (2002) define que “[...] no chamado CRM o objetivo não é apenas conhecer melhor o cliente, mas principalmente, buscar de forma incessante o atendimento de suas necessidades de forma que a satisfação desse cliente se transforme em fidelidade à marca”.

Um meio de utilizar o CRM na Internet é através da consulta personalizada ao *site*, seja esta consulta com ou sem o cliente identificado pelo sistema, onde o marketing 1-1 (*one-to-one*), típico do CRM, deve ser aplicado.

Agentes de *interface* enfatizam a autonomia e a aprendizagem para resolução das tarefas empregadas pelos seus proprietários. Essencialmente, os agentes de *interface* suportam e fornecem o auxílio pró-ativo a um usuário para aprende usar uma aplicação particular ou rotina em específico. O agente de *interface* observa as ações realizadas pelo usuário no estilo “presta atenção sobre o ombro de seu usuário”, e sugerem maneiras melhores de fazer a tarefa (Jennings e Woodridge 1998).

Baseando-se neste cenário, com a aplicação deste trabalho, foram identificadas as informações dos usuários, visitantes não “logados” ou clientes já “logados” da aplicação, para isto, no campo consulta do site na página principal, há o armazenamento na base de dados de todos os produtos consultados que constam cadastrados no sistema, ou seja, caso haja o produto na base de dados, este será armazenado como consultado, caso seja consultado pelo cliente identificado, esta consulta será armazenada também com sua identificação.

Neste trabalho utilizou-se agentes de *interface*, o primeiro, que é chamado de Agente de *Interface* de Consulta (AIC), utilizará a base de consulta e a base das locações de todos os clientes para selecionar alguns produtos, para que após, estes possam ser exibidos quando a página for carregada pelo *browser*, exibindo assim os produtos com os parâmetros iguais aos mais consultados e locados pelos clientes, com isso simulando o CRM.

Um usuário ao visitar um *site* buscando por qualquer tipo de informação ou serviço, defronta-se com a generalização de visitantes, generalização essa que não fornece qualquer tipo de personalização por parte do *site* para um usuário em específico.

Para a solução deste problema de personalização, outro agente é utilizado, este chamado de Agente de *Interface* de Indicação (AII). Onde o usuário já cadastrado realiza o

*login* na aplicação para identificar-se, ativando o AII. Este agente verifica os históricos de locações e consultas realizadas anteriormente pelo cliente, comunica-se com o AIC para saber quais os produtos indicados por ele, e indica produtos de acordo com seu perfil e que possam ser de sua preferência.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é desenvolver uma aplicação utilizando agentes de *interface* para personalizar a consulta e indicar produtos de acordo com o perfil dos clientes da aplicação.

Os objetivos específicos do trabalho são:

- a) Identificar o perfil do usuário que utiliza a aplicação através de operações que retém informação sobre as ações do mesmo;
- b) Disponibilizar para o cliente produtos direcionados ao seu perfil através do AII;
- c) Identificar através dos AIC os parâmetros dos produtos mais consultados e de maior circulação.

## 1.2 ESTRUTURA DO TRABALHO

O trabalho foi desenvolvido observando várias etapas. No capítulo 1 foi apresentada a introdução ao trabalho desenvolvido, bem como seus objetivos e sua organização.

O segundo capítulo define informação e a geração de conhecimento, descreve agentes e agentes de *interface*. Também contextualiza o sistema de informação Customer Relationship Management (CRM), sua aplicação e os resultados que podem ser atingidos utilizando o mesmo.

O terceiro capítulo apresenta a análise, casos de uso e diagramas referente ao sistema implementado, descreve as tecnologias utilizadas e a implementação do sistema assim como o resultado de sua utilização.

No capítulo 4 são apresentadas as conclusões do trabalho e sugestões para trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos utilizados no desenvolvimento da aplicação, sendo eles a obtenção de informação e o conhecimento, os agentes de *interface* e CRM.

### 2.1 INFORMAÇÃO E CONHECIMENTO

Weitzen (1991) utiliza o exemplo da Curva de Má Distribuição de Pareto, hoje conhecida como *Regra dos 80/20*, para demonstrar a necessidade da obtenção correta das informações pela organização, onde o economista Wilfredo Pareto, por volta de 1800 descobriu que em diversas companhias 20% dos produtos são responsáveis por 80% das vendas. Pareto também verificou que 20% dos clientes geram 80% da renda. O princípio de Pareto pode ser demonstrado no contexto do gerenciamento de informações, onde 20% das informações contribuem diretamente para 80% das decisões de uma organização.

Carvalho e Tavares (2001) definem que a informação essencial é aquela que contribui para a organização e apresenta-se de forma limpa sistematizada e racional. Sendo que através da informação, ou melhor, da soma correta das informações, é formado o conhecimento.

Aranha e Martins (1998) atribuem a descrição de conhecimento ao ato, o processo pelo qual o sujeito se coloca no mundo e com ele estabelece uma ligação, ligação esta estabelecida pelas informações e da forma pela qual elas lhe são apresentadas.

Do ponto de vista das organizações. Nonaka e Takeuchi (1997), citam um importante contraste, de como a forma correta da utilização do conhecimento nas empresas orientais formam empresas competitivas e preparadas para o mercado ocidental. As empresas orientais após a Segunda Guerra Mundial passaram por vários períodos conturbados no mercado que

estão inseridos. Nestes cenários, as informações, como a obtida por seus fornecedores, clientes e até concorrentes, são transformado em conhecimento e compartilhado de forma ampla dentro da organização, armazenando como parte da base de conhecimentos da empresa e utilizado pelos envolvidos no desenvolvimento de novas tecnologias, produtos e serviços.

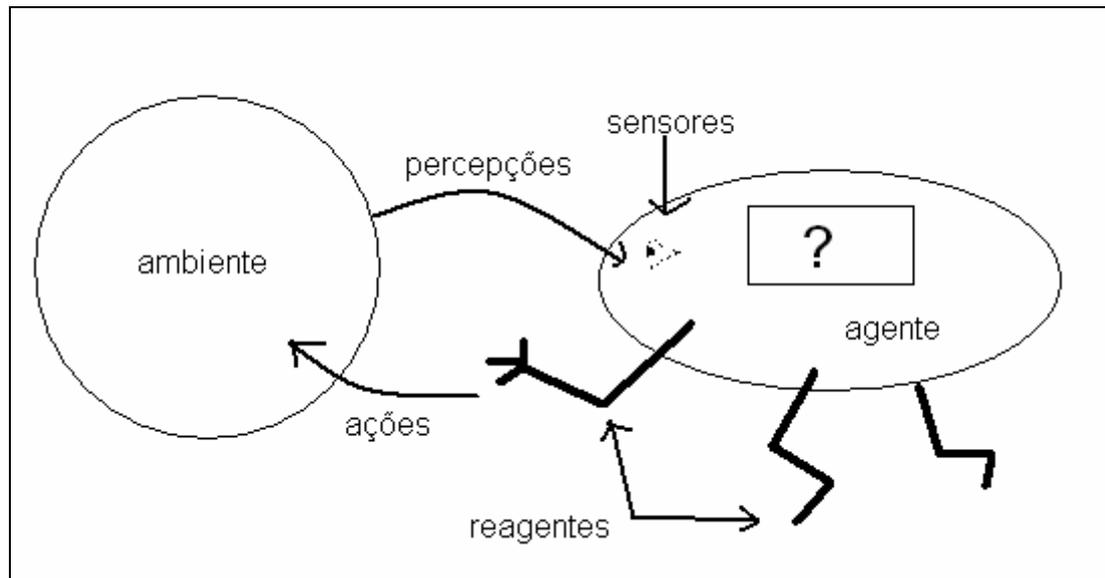
## 2.2 AGENTES DE SOFTWARE

Para Bre (1998 apud FRAIZ, 2000), não é possível ainda aceitar um conceito genérico de Agentes. Isto está evidenciado pela característica interdisciplinar dos agentes, no qual uma corrente está sujeita aos efeitos de diferentes áreas científicas e, outra corrente reflete a demanda requerida das aplicações práticas. Se existe aproximação da área de agentes fundamentalmente na direção da Inteligência Artificial, há diferentes critérios daqueles que vêem a informação e comunicação de sistemas. Similarmente, a demanda de cientistas sociais é diferente daqueles mais orientados a observações técnicas feitas por cientistas da computação. Mas uma definição que vem sendo disseminada entre os estudiosos da área é a de Huhns e Singh (1998) que especifica que os agentes são componentes ativos, persistentes (do software) que percebem, raciocinam, agem, e comunicam-se com outros e agentes e/ou outras aplicações afim de encontrar a solução para o problema em questão.

Seguindo este conceito pode-se dizer que um agente pode ser definido como alguém ou alguma coisa que atua como um representante de outrem, com o propósito de desempenhar ações que são benéficas para ao solicitante (Faraco 1998). E também que um agente representa o usuário, ajuda o usuário, guia o usuário, e em alguns casos, toma ação unilateral no interesse do usuário (Bigus e Bigus 1997).

A figura 1 apresenta uma definição de agentes, que para Bogo (2000), um agente é descrito como um programa de software que auxilia o usuário na realização de alguma tarefa

ou atividade.



Fonte: Adaptado de Russel e Norvis (1995)

Figura 1 : Definição de agentes

Os agentes tratados neste trabalho são os agentes de *interface*, onde são utilizados de forma a auxiliar o usuário em sua navegação.

### 2.2.1 AGENTES DE INTERFACE

Para Huhns e Singh (1998), os agentes de interface são os sistemas semi-inteligentes que ajudam os usuários com tarefas rotineiras, tais agentes aprendem prestando atenção por cima do ombro o comportamento do usuário. Ou seja, possuem técnicas da aprendizagem empregadas que têm como objetivo fornecer o auxílio a um usuário em uma aplicação em específico.

Bradshaw (1997) define que os agentes de *Interface* não agem como uma relação ou uma camada entre o usuário e a aplicação. Comporta-se como um assistente pessoal que coopera com o usuário na tarefa que lhe foi atribuída.

Agentes de interface possuem diferentes designações:

- Escondem a complexidade de tarefas difíceis;
- Executam tarefas no interesse do usuário;
- Podem treinar ou ensinar o usuário;
- Ajudam a usuários diferentes collaborate;
- Monitoram eventos e procedimentos.

As tarefas nas aplicações em que um agente está inserido e que podem ajudar o usuário são virtualmente ilimitadas: filtrar informações, recuperar informação, gerência do correio, seleção dos livros, filmes, música, etc. (Bradshaw 1997).

Quanto a aprender, tipicamente, agentes de *interface* podem aprender de quatro maneiras:

1. Observando e imitando o usuário;
2. Reagindo ao *feedback* positivo ou negativo do usuário;
3. Recebendo instruções explícitas do usuário;
4. Pedindo conselho para outros agentes.

Geralmente, o modo de aprendizagem é dado pela consulta a dados ou parâmetros armazenados em memória, embora outras técnicas, tais como a aprendizagem evolucionária, também fazem parte de seu escopo. Cooperação com outros agentes, se existentes, são limitadas a pedir conselho (Jennings e Wooldridge 1998).

### 2.3 CUSTOMER RELANTIONSHIP MANAGEMENT (CRM)

Neste capítulo é apresentada uma breve introdução sobre o CRM em aplicações comerciais, seguido do conceito, algumas aplicações e o retorno da utilização deste ramo de sistemas de informação.

### 2.3.1 INTRODUÇÃO

Swift (2001) define a Gerência de Relacionamento com o Cliente (CRM), como uma abordagem empresarial destinada a entender e influenciar o comportamento dos clientes, por meio de comunicações para melhorar a retenção e a lealdade deles. A palavra importante é relacionamento, onde o foco da empresa deve ser voltado ao cliente e sua satisfação.

Conforme Cardoso e Gonçalves Filho (2001), é possível identificar que quase 70% das razões pelas quais os clientes abandonavam a marca estavam ligadas a atendimento e falta de atenção dada aos clientes. Apenas 15% deles mudaram para um competidor por ter encontrado um produto melhor e 15% por ter encontrado um produto mais barato.

Assim, utilizando as informações sobre o cliente e um atendimento personalizado para gerar lucros para a empresa.

### 2.3.2 CONCEITO

Para Felipini (2002), uma questão central no CRM é fazer com que cada contato sirva como um aprendizado sobre o cliente de forma a conhecer cada vez mais suas características e suas necessidades.

Segundo Souza (2004), o CRM tem essencialmente o foco em obter o valor máximo dos clientes, seja através da maneira que as companhias se comunicam com eles, como vendem, e como lhes prestam serviços de atendimento ou manutenção, como também pelos meios tradicionais do produto, do preço, da promoção e do ponto de venda.

Com um conhecimento mais aprofundado dos consumidores obtido através da interatividade, é possível saber tanto sobre os seus clientes, que o produto adapta-se às suas necessidades de tal forma que quase não é necessário fazer mais nada para convencer o cliente

a adquirir os produtos ou serviços da empresa. (Hanson, 2003)

A seguir, são citadas empresas que optaram por utilizar o CRM como um de seus diferenciais.

### 2.3.3 APLICAÇÃO DO CRM

O CRM está sendo utilizado por várias grandes empresas, dentre elas Amazon.com e Telefônica , e sendo criado por outras diversas, dentre as principais destacam-se a Microsoft, Siebel e SAP (Hanson, 2003).

A Amazon.com, optou por um CRM onde, segundo Felipini (2002), o cliente ao registrar seu nome, cada vez que retornar ao *site*, vai ser recebido com uma saudação personalizada. Se o mesmo comprar um livro de informática, e se autorizar, vai ser sempre avisado sobre novos lançamentos e promoções na área, desta forma passando a impressão ao cliente que há um foco no que ele está fazendo e um interesse por parte da empresa em mantê-lo fiel.

Bottoni e Zocchio (2006) cita o caso da empresa Network1 onde a empresa diferenciou o atendimento indo pelo caminho mais difícil e rompendo a barreira da impessoalidade entre vendedor e cliente. Onde a empresa passou a ligar para os clientes não para oferecer seus produtos, mas por outros motivos, como dar parabéns quando o time para o qual torcem vence um campeonato. Assim o relacionamento passou a ser mais prazeroso e não só comercial e o cliente satisfaz-se com o atendimento.

### 2.3.4 RETORNO DO USO DO CRM

Uma empresa, assim como qualquer organização com fins lucrativos, visa receber

lucro com toda ação tomada em seu negócio.

Para Souza (2004), acompanhamento da lucratividade dos clientes é fundamental para o estabelecimento de estratégias de diferenciação. Os executivos esperam cada vez mais acompanhar a lucratividade de seus clientes e ter esses dados disponíveis. Se a organização por tipos de clientes os diferencia por suas necessidades, é acompanhamento e a aferição da lucratividade que os permite diferenciá-los por valor.

Segundo Bottoni e Zocchio (2006), o investimento no CRM para a empresa Network1 resultou num retorno considerável, onde as vendas aumentaram cerca de 20% no tíquete médio de vendas.

### 3 DESENVOLVIMENTO

Neste capítulo é apresentado o desenvolvimento da aplicação utilizada neste trabalho, assim a análise do sistema e a apresentação da aplicação em si.

#### 3.1 ESPECIFICAÇÃO

Foi levantada na empresa a ausência de meios para manter os clientes, sendo que não é investido em marketing externo, apenas é focado o atendimento e divulgação da empresa no próprio estabelecimento e no popular “boca a boca”. Foi então optado por utilizar o CRM como forma de marketing para manter o cliente fiel à empresa, passando a ele a impressão de ser único e vital para a empresa.

Para realizar a implantação de um CRM na JF Vídeo Locadora foi necessário verificar que tipo de diferenciação seria utilizado, optou-se por recomendar títulos aos clientes utilizando agentes de *interface* através de uma aplicação onde seriam armazenados alguns dados durante a permanência do cliente no mesmo.

#### 3.2 PROJETO

A aplicação utiliza dois agentes de *interface* chamados de Agente de *Interface* de Consulta (AIC) e Agente de *Interface* de Indicação (AII). Estes buscam na base de dados as locações e outros dados referentes ao cliente e com isso recomendam títulos que podem ser da preferência do cliente, são caracterizados como agentes de *interface* pelo fato de que, através de reagentes de suas ações (*feedback*), tratados na seção 3.3, o AII verifica se suas indicações estão afetando realmente o cliente e o mesmo está optando por elas.

### 3.2.1 REQUISITOS

No quadro 1 são apresentados os requisitos funcionais levantados para o sistema.

Requisitos Funcionais	Caso de Uso
RF01: O sistema deverá permitir ao usuário a alteração de seu <i>login</i> e senha.	UC01
RF02: O sistema deverá permitir ao usuário consultar títulos no estoque dividido por gêneros específicos.	UC02
RF03: O sistema deverá permitir ao usuário reservar de títulos em estoque para a data desejada.	UC03
RF04: O sistema deverá permitir ao AIC indicar títulos com os parâmetros dos filmes mais e locados e mais consultados, na tela principal do sistema.	UC04
RF05: O sistema deverá permitir ao AII sugerir, através de uma janela, ao usuário identificado, produtos que poderiam ser de sua preferência.	UC05
RF06: O sistema deverá permitir ao usuário buscar filmes em estoque.	UC06
RF07: O sistema deverá permitir ao usuário consultar os 10 títulos mais locados.	UC07
RF08: O sistema deverá armazenar a consulta do usuário, através de <i>clickstream</i> .	UC08
RF09: O sistema deverá permitir ao usuário consultar sobre futuros lançamentos.	UC9
RF10: O sistema deverá permitir ao usuário a configuração de suas preferências primárias.	UC10
RF11: O sistema deverá permitir ao usuário a consulta de seu histórico de locações.	UC11

Quadro 1 – Requisitos funcionais

O quadro 2 apresenta os requisitos não funcionais do sistema, estes requisitos poderiam ser ou não implementados.

<b>Requisitos Não Funcionais</b>
RNF01: O sistema deverá ser implementado em PHP 5 com banco de dados Interbase 7.
RNF02: O sistema deverá interagir com o sistema para vídeo locadoras, já implantado na JF vídeo locadora e <i>games</i> Ltda me.
RNF03: O banco de dados deverá ser utilizado por ambas as aplicações, tanto <i>online</i> pela aplicação <i>web</i> como <i>offline</i> pela aplicação na empresa.
RNF04: Os agentes deverão ser implementados com regras de seleção diferenciadas

Quadro 2 –Requisitos não funcionais

### 3.2.2 CASOS DE USO

A figura 2 apresenta os casos de uso relacionados às ações referentes que o usuário pode desempenhar dentro da aplicação. Este diagrama demonstra a relação existente entre os agentes e as interações com os usuários.

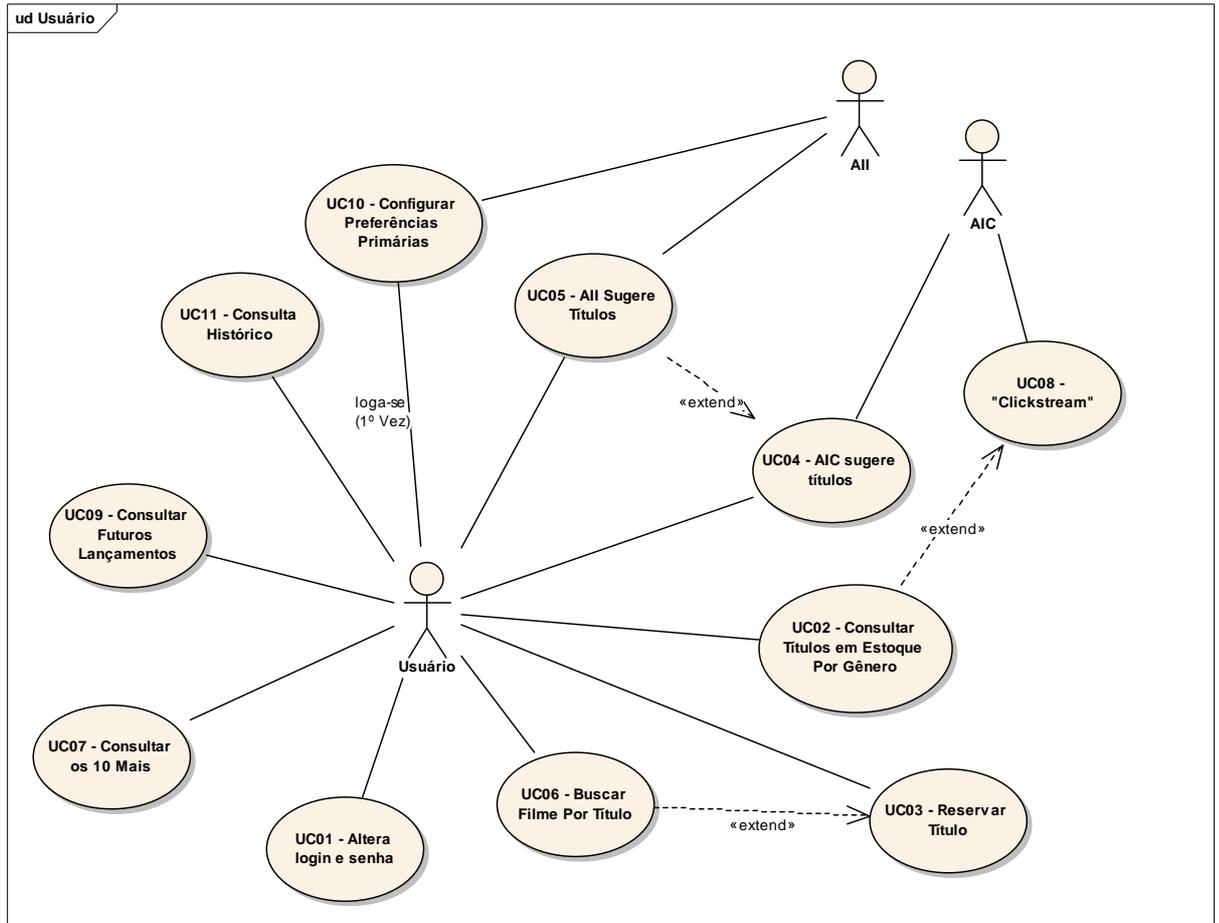


Figura 2 – Diagrama de caso de uso do usuário

### 3.2.3 DIAGRAMA DE CLASSES

A figura 3 apresenta o diagrama de classes utilizadas na aplicação, sendo ela implementadas na linguagem PHP.

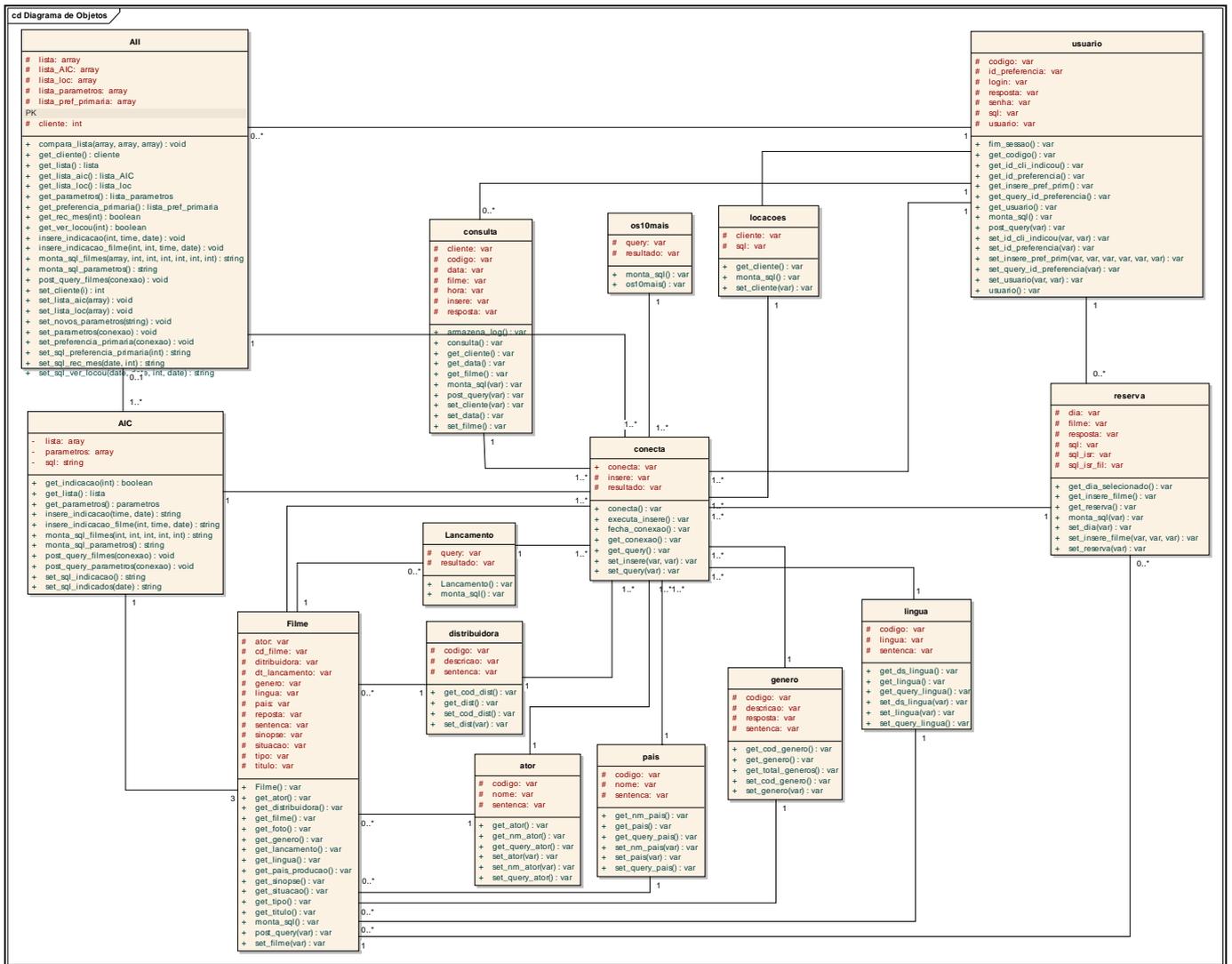


Figura 3 –Diagrama de classe

A seguir, no quadro 3, é apresentada a descrição de cada classe do diagrama de classes da figura 3.

Através deste detalhamento torna-se mais fácil a compreensão do sistema e a demonstração de como cada classe age dentro da aplicação.

<b>Classe</b>	<b>Descrição</b>
Conecta	Classe que realiza a conexão com o banco de dados da aplicação, possui métodos que conectam com o banco e que realizam a execução das queries de consulta.
AIC	Contém os atributos e métodos comuns ao AIC, sendo visualizada por qualquer usuário que utiliza a aplicação.
AII	Contém os atributos e métodos comuns ao AII, é utilizada apenas por usuários cadastrados e <i>logados</i> no sistema.
Ator	Contém os atributos e métodos dos atores, utilizados pela classe Filme para obter os nomes dos atores dos títulos.
Consulta	Classe que contém atributos e métodos que realizam a função de <i>clickstream</i> , armazenando as consultas realizadas pelos clientes <i>logados</i> ou não no sistema.
Distribuidora	Classe contém atributos e métodos das distribuidoras dos títulos, utilizada pela classe Filme para obter o nome das distribuidoras.
Filme	Classe que utiliza seus métodos para obter informações dos títulos.
Gênero	Contém os atributos e métodos dos gêneros, utilizados pela classe Filme para obter a descrição do gênero do título.
Lançamento	Classe possui atributos e métodos utilizados para obter os próximos lançamentos.
Língua	Contém os atributos e métodos dos Idiomas, utilizados pela classe Filme para obter a descrição dos idiomas dublados dos títulos.
Loações	Classe que contém os atributos e métodos utilizados para obter as locações do cliente armazenadas na Base de Dados
Os10Mais	Classe contém atributos e métodos para mostrar os 10 Filmes mais Locados no Total.
País	Contém os atributos e métodos dos Países cadastrados para a aplicação, utilizados pela classe Filme para obter o nome do país de produção dos títulos.
Reserva	Classe que possui atributos e métodos para consultar e efetuar reservas para os clientes <i>logados</i> .
Usuário	Classe que contém atributos e métodos referentes ao cliente <i>logado</i> na aplicação, utilizada para verificar <i>login</i> e senha.

Quadro 3 – Descrição das classes utilizadas na aplicação

### 3.2.4 DIAGRAMA FÍSICO DE DADOS

Nas figuras 4 e 5 é apresentado o modelo físico entidade-relacionamento (MER) da base de dados utilizada pela aplicação como forma de armazenamento de dados.

A base de dados foi implementada utilizando o bando de dados Interbase 7, banco este escolhido por já ser utilizado na aplicação local.

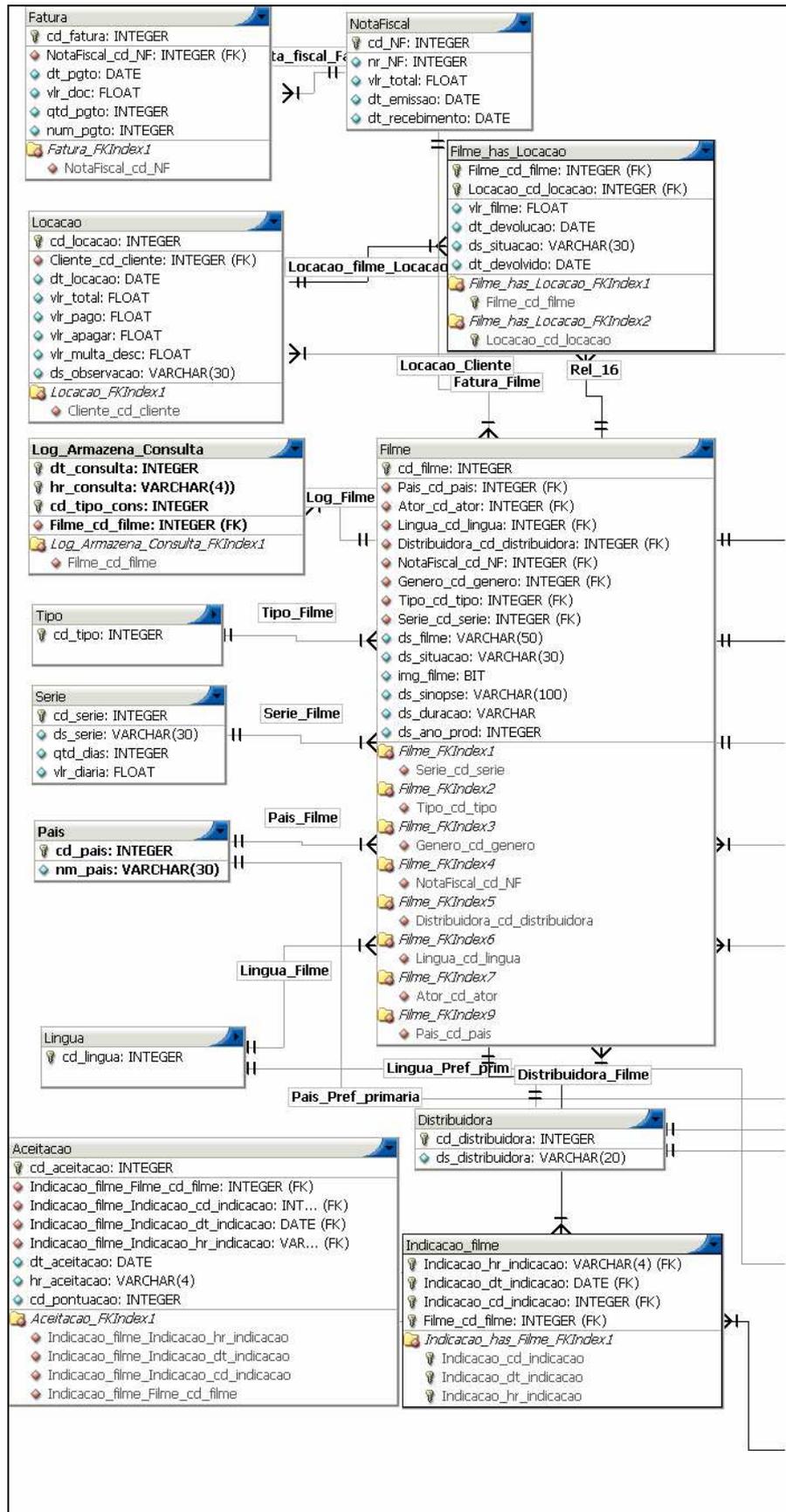


Figura 4 – Modelo físico de dados

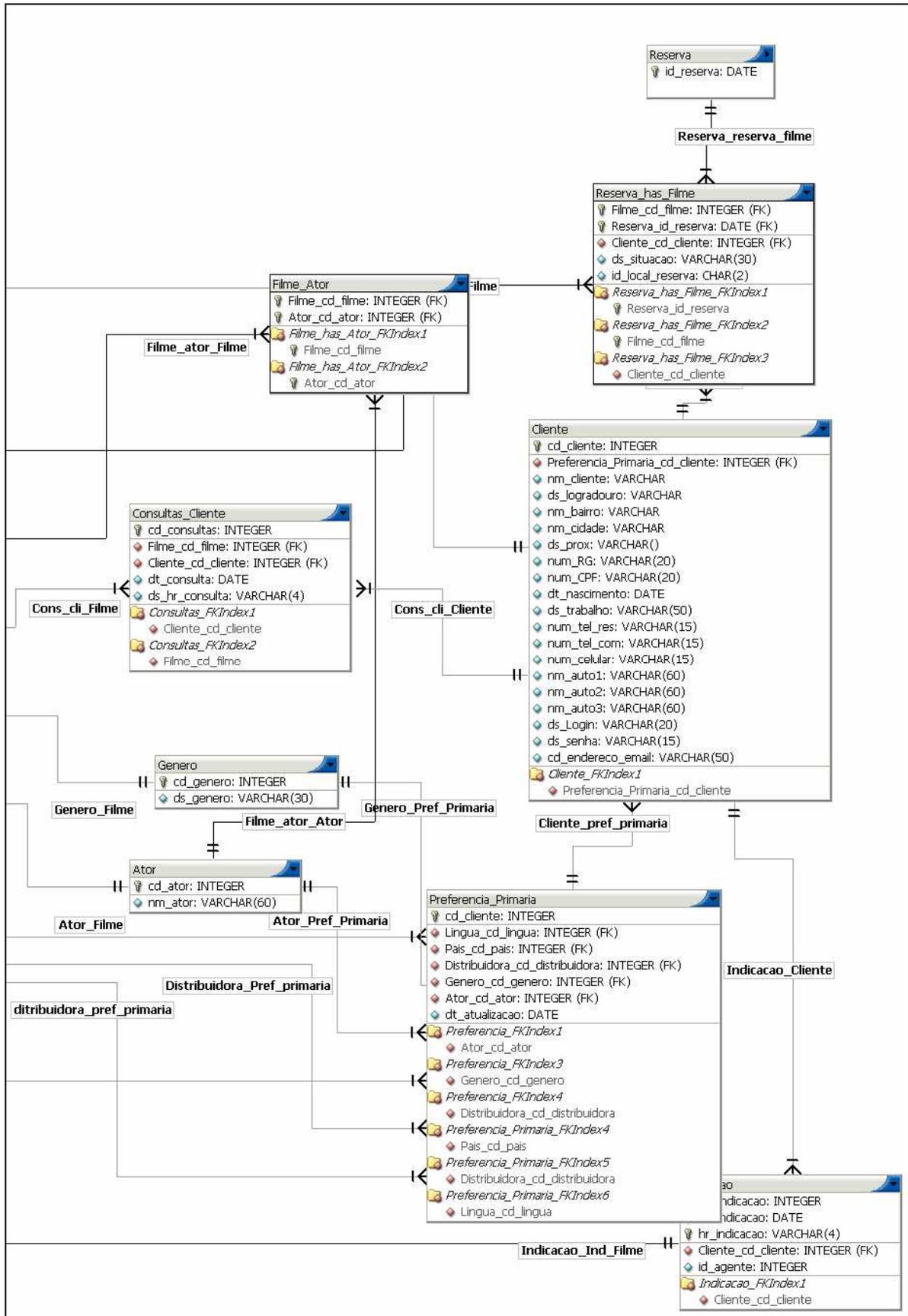


Figura 5 – Modelo físico de dados (continuação)

### 3.2.5 DIAGRAMA DE ATIVIDADES

Na figura 6 é apresentada o diagrama de atividades a partir do momento que o cliente *loga-se* até o momento que reserva um título consultado por ele, é demonstrado também a atuação do AIC e do AII na recomendação de títulos ao cliente.

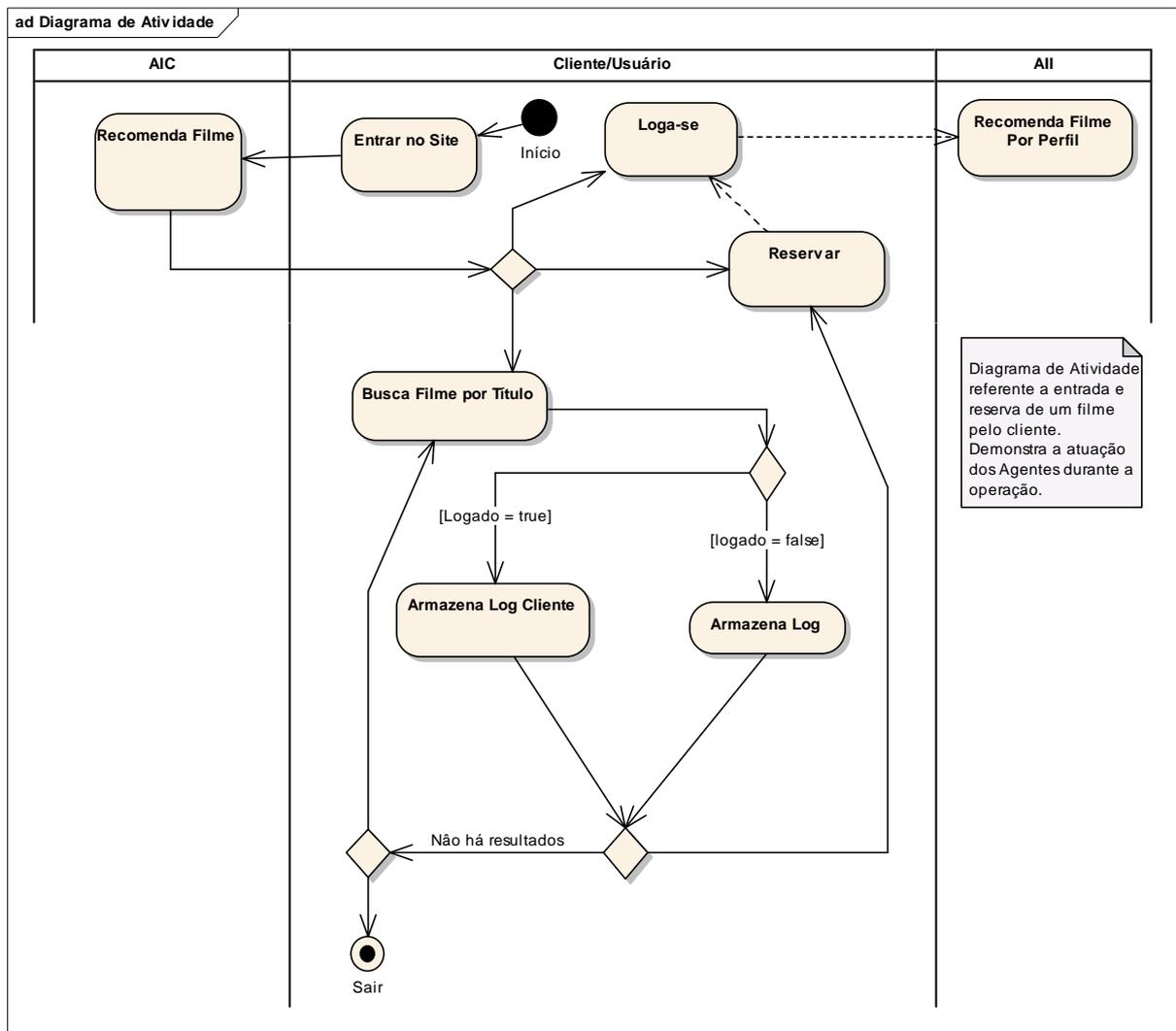


Figura 6 – diagrama de atividades

### 3.3 TECNOLOGIAS E FERRAMENTAS UTILIZADAS

Neste capítulo são apresentadas as ferramentas e tecnologia utilizada para o desenvolvimento do trabalho.

#### 3.3.1 PHP

Soares (2000) explica o PHP como uma combinação de linguagem de programação e servidor de aplicações, onde pode-se programar em PHP como em qualquer outra linguagem, mas o que difere o PHP das outras linguagens é sua capacidade de interagir com o mundo WEB.

Para Gerken e Ratschiller (2000) o PHP não é uma linguagem em si, mas uma mistura de linguagens diferentes, tendo o C como a linguagem de programação que mais assemelha-se ao seu desenvolvimento.

O PHP está na sua quinta versão atualmente onde, é considerada por Bakken e Rethans (2005) como a versão mais consistente.

### 3.4 IMPLEMENTAÇÃO DA APLICAÇÃO

A aplicação foi implementada utilizando a linguagem de programação PHP, que segundo Bakken e Rethans (2005) é a linguagem mais utilizada na internet atualmente, e banco de dados Interbase 7.0, que foi escolhido por já ser utilizado na aplicação local da empresa.

Optou-se por uma abordagem de criação de classes por haver a possibilidade de interação das classes e a fácil utilização de seus métodos. A primeira classe implementada é a

Conexão, esta classe faz a conexão do PHP com o banco de dados, esta conexão é realizada pelo método “conecta()”, que funciona como o construtor da classe e é acionado ao ser criado o objeto conecta, a classe possui ainda dois métodos muito importantes, o método “fecha\_conexao()”, o método “get\_conexao()” é utilizado pelo PHP como parâmetro ao realizar as *queries* de consulta e inclusão no banco.. A figura 7 apresenta os métodos supracitados.

```

Class conecta {

public $conecta;
protected $resultado;
protected $insere;

public function conecta(){
    //Conecta com o Banco

    $host = "localhost:c:/Paginas/bd.gdb";
    $username = "SYSDBA";
    $password = "masterkey";
    // conecta com o banco
    $conexao = ibase_connect($host,$username,$password);
    $this->conecta = $conexao;
} // Fim da função

public function get_conexao(){
    return $this->conecta;
}

public function fecha_conexao(){
    ibase_close($this->conecta);
}
}

```

Figura 7 – métodos da classe Conecta

A classe Gênero utiliza o método “set\_cod\_genero()” para obter os dados dos gêneros utilizando o código obtido através da função “set\_genero(\$cod)”, onde *\$cod* é o código passado como parâmetro e que atribui o valor passado ao atributo da base *\$código*. A classe Gênero, assim como seus atributos e métodos, são apresentados na figura 8.

As classes Ator, Língua, País, Distribuidora utilizam o mesmo princípio da classe Gênero para consultar e obter resultados nas consultas.

```

// classe de gêneros
class genero{
protected $codigo;
protected $descricao;
protected $sentenca;
protected $resposta;

public function set_genero($cod){
    $this->codigo = $cod;
}

// Funcao que monta o sql para consulta dos codigos
// exclui códigos que não são de filmes
public function set_cod_genero(){
    $this->sentenca = 'select
        cd_genero,
        ds_genero
    from
        genero
    where
        cd_genero not in (19,0)
    order by ds_genero';
}

// retorna o sql de consulta de generos
public function get_cod_genero(){
    return $this->sentenca;
}

public function get_genero(){
    return $this->codigo;
}

```

Figura 8 – Classe Gênero

As principais classes utilizadas na implementação são, indubitavelmente, as classes dos agentes, AII e AIC.

A classe AIC, utiliza métodos que executam *queries* no banco de dados para obter os parâmetros dos títulos mais consultados e mais locados de todos os clientes que navegam na aplicação, sem *logar-se*, e que locam ou já locaram no estabelecimento. O método “*monta\_sql\_parametros()*” contém o comando SQL que será utilizado pelo método “*post\_query\_parametros(\$resp)*”, onde *\$resp* é passado como parâmetro o método “*get\_conexão()*” da classe Conecta, para navegar entre os resultados obtidos após a consulta no banco de dados.

Na figura 9 são apresentados os métodos responsáveis por consultar títulos que possuem os parâmetros comuns obtidos pelos métodos supracitados, a função “*get\_lista()*” é utilizada pelo AII para obter a lista de títulos indicados pelo AIC.

```

// Monta query de consulta dos filmes que se encaixam do perfil
// (parâmetros) passado
public function monta_sql_filmes($gen, $ator, $dist, $lingua, $pais){
    $this->sql = 'select
        cd_filme
    from
        filme
    where
        cd_genero = '.$gen.'and
        cd_ator_principal IN ( '.$ator.' )and
        cd_dist = '.$dist.'and
        cd_lingua = '.$lingua.'and
        Cd_pais_producao = '.$pais.' and
        ds_filme not in (select
            f.ds_filme
        from
            indicacao_filme i,
            filme f
        where
            f.cd_filme = i.cd_filme)';

    return $this->sql;
}
// É necessário passar o resultado da conexão
public function post_query_filmes($resp){
    $this->lista = ibase_fetch_row($resp);
}

// retorna array com os dados
public function get_lista(){
    return $this->lista;
}

```

Figura 9 – métodos da classe AIC

Os métodos “set\_sql\_indicação(\$data)”, onde *\$data* é a data atual passada como parâmetro, e “get\_indicacao(\$ok)”, onde *\$ok*, são utilizados para conferir se o AIC já realizou alguma indicação no dia.

Os métodos “insere\_indicacao(\$data, \$hr)”, onde *\$data* e *\$hr* são parâmetros de datas, e “insere\_indicacao\_filmes(\$data, \$hr, \$filme)”, onde *\$filme* é o código do filme passado como parâmetro, são para inclusão dos dados nas tabelas “Indicacao” e “Indicacao\_Filme”, tabelas estas que armazenam as indicações já realizadas pelo AIC.

O método “set\_sql\_consulta(\$data)”, onde *\$data* é a data atual passada como parâmetro, é utilizado para consultar os filmes que já foram indicados na data passada como parâmetro.

A classe AII, correspondente ao Agente de *Interface* de Indicação, contém atributos que são “setados” através de métodos de outras classes, como por exemplo, o atributo *\$cliente* que armazena o código do cliente *logado* é “setado” através do método *set\_cliente(\$cli)* onde *\$cli* é o código do cliente obtido através do método *get\_cliente()* da classe Usuário.

A seguir são apresentados os métodos mais importantes da classe AII, sendo eles responsáveis por avaliar as indicações do agente.

#### 3.4.1 ATUAÇÃO DO AGENTE AII

O AII contém alguns métodos que são responsáveis em avaliar o *feedback* do cliente às indicações do agente. Onde cada vez que o agente é ativado, ao passar um mês de uma indicação, ou seja, se sua primeira indicação, o AII verifica se o cliente está gostando dos parâmetros que o agente está utilizando para indicação e caso o retorno for negativo, utiliza outros parâmetros para recomendação.

Os métodos *set\_preferencia\_primaria(\$resp)* onde *\$resp* é um parâmetro passado o resultado da consulta no bando de dados, e *get\_preferencia\_primaria()*, fazem a consulta na tabela “*Preferencia\_primaria*”, onde estão armazenadas as preferências primárias indicadas pelo cliente na primeira que ele *loga-se* na aplicação.

O método *set\_sql\_rec\_mes(\$cli, \$data)*, onde *\$cli* é o código do cliente e *\$data* o dia atual atrasado em um mês, junto com os métodos *set\_sql\_rec\_mes(\$ok)* e *set\_sql\_ver\_locou(\$data, \$cli, \$data)* e *get\_ver\_locou(\$ok)*, servem como o sensor do agente, que verifica se suas indicações dirigidas a este cliente estão surtindo efeito, caso não estejam, o método *set\_novos\_parametros(\$sql)*, apresentado na figura 10, obtém novos parâmetros para realizar novas indicações ao cliente.

No capítulo 6, são apresentados os códigos-fonte das duas classes, AII e AIC.

```

// se o cliente não locou recentemente nenhum filme indicado pelo agente
// o agente utiliza resultado com os parâmetros obtidos
// pela query de consulta de perfil
// com o outros parâmetros
// ----- Deve passar a conexao -----
Public function set_novos_parametros($sql){
    // deve procurar os parâmetros de preferência de segunda opção
(segunda linha da query)
    ibase_fetch_row($sql);
    // armazena o resultado
    // com os parâmetros preferenciais
    $this->lista_parametros = ibase_fetch_row($sql);
}

```

Figura 10 – método set\_novos\_parametros(\$sql)

### 3.5 APRESENTAÇÃO DA APLICAÇÃO

A seguir são apresentadas as telas da aplicação onde, primeiramente dá-se a tela principal.

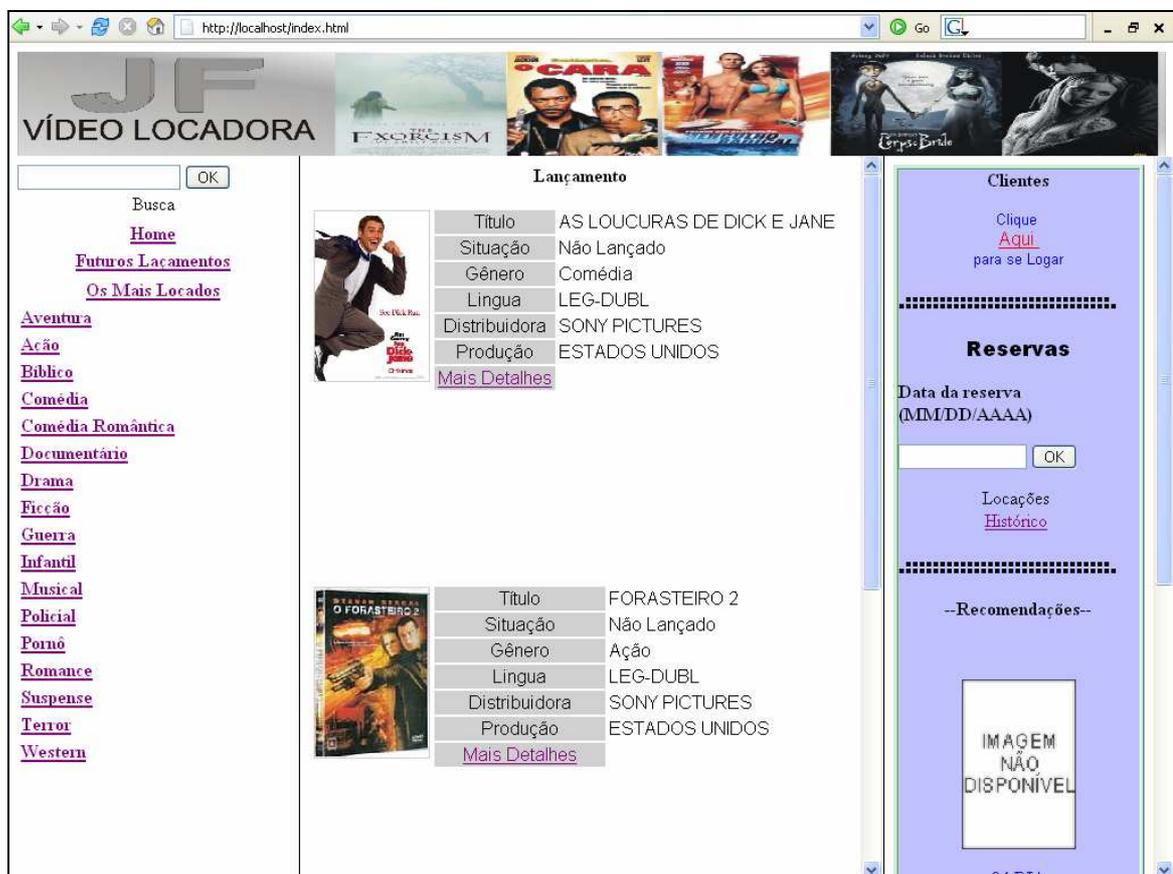


Figura 11 – Página inicial da aplicação

A figura 11 apresenta a tela inicial da aplicação, o menu na figura 12 é utilizado para a navegação pela aplicação.



Figura 12– Menu principal

Através desse menu ocorrem às consultas pelos títulos na base de dados, é possível se pesquisar por um filme digitando o título do mesmo no campo “Busca” do menu e clicando em “Ok” onde o filme será mostrado na tabela do meio, como mostra a figura 13.



Figura 13 – Consulta por títulos dos filmes

Com isto, se houver o filme em estoque, assim como ocorreu na figura 14, o sistema armazenará o código do filme, a hora e data da consulta, para ser utilizado posteriormente pelos agentes.

A figura 14 apresenta a consulta de futuros lançamentos solicitada com o clique na opção logo a baixo da supracitada, onde, será mostrados os títulos próximos de chegar no estabelecimento.

The screenshot shows a web browser window with the URL `http://localhost/index_2.html`. The page title is "TCC Jonathan Francesco Steilein" and the search bar contains "espíritos dvd - Pesquisa Google". The main content area displays two movie listings:

	Título	NO RASTRO DA BALA
	Situação	Não Lançado
	Gênero	Ação
	Lingua	LEG-DUBL
	Distribuidora	IMAGEM FILMES
	Produção	ESTADOS UNIDOS
	Data de Lançamento	2006-06-22
		<a href="#">Mais Detalhes</a>

	Título	ZATHURA UMA AVENTURA EPICAL
	Situação	Não Lançado
	Gênero	Aventura
	Lingua	LEG-DUBL
	Distribuidora	SONY PICTURES
	Produção	ESTADOS UNIDOS
	Data de Lançamento	2006-06-28
		<a href="#">Mais Detalhes</a>

On the right side, the "Recomendações" section displays three movie posters with their respective titles: "24 DIA", "7 SEGUNDOS", and "O NOVO".

Figura 14 – Futuros lançamentos

Os títulos mais locados são mostrados na figura 15 através de um clique com o *mouse* na opção os “Os Mais Locados” do menu. Nesta tela de títulos mais locados são apresentados os 10 títulos mais locados no comércio.

Os 10 Mais Locados		
Posição	Título	Detalhe
1°	FERAS DO ASFALTO	<a href="#">Vizualizar</a>
2°	HERBIE MEU FUSCA TURBINADO	<a href="#">Vizualizar</a>
3°	SR E SRA SMITH	<a href="#">Vizualizar</a>
4°	O QUARTETO FANTASTICO	<a href="#">Vizualizar</a>
5°	AMERICAN PIE 4	<a href="#">Vizualizar</a>
6°	O FILHO DO MASCARA	<a href="#">Vizualizar</a>
7°	GOLPE BAIXO	<a href="#">Vizualizar</a>
8°	O VIRGEM DE 40 ANOS	<a href="#">Vizualizar</a>
9°	SIN CITY	<a href="#">Vizualizar</a>
10°	O ATIRADOR - APONTANDO PARA A MORTE	<a href="#">Vizualizar</a>

Figura 15 – Os mais locados

O menu da direita apresentado na figura 16, este é direcionado aos clientes da aplicação, que possuem um *login* e uma senha válidos e assim poderão navegar pela aplicação livremente, este menu também apresenta as recomendações do AIC na parte inferior do mesmo.

**Cientes**

Clique  
[Aqui](#)  
para se Logar

---

**Reservas**

Data da reserva  
(MM/DD/AAAA)

Locações  
[Histórico](#)

---

--Recomendações--



24 DIA

Figura 16 – Menu de navegação do cliente

O cliente ao se *logar* através da opção “Clique aqui para se Logar” lhe é solicitado o *login* e a senha fornecidos previamente no estabelecimento, através do sistema atual já implantado que gera diretamente para impressão ao cliente o cartão exibido na figura 17, e caso for a primeira vez que está acontecendo o acesso a tela da figura 18 lhe é oferecida, onde através dela é selecionada uma nova senha e as preferências primárias do cliente, fundamental para a recomendação do AII.

<b>JF Vídeo Locadora E Games Ltda Me.</b>	
Rua: Professor Max Humpl 2391 - Salto do Norte Blumenau - SC	Tel.:3334-0464
<u>Senha e Login Para Uso no Nosso Portal</u>	
Cliente	<b>HANS ROBERT WINKLER</b>
Login:	<b>HANS</b>
Senha:	1535315

Figura 17– Cartão com senha do cliente

Cliente  
**HANS ROBERT WINKLER**  
 Logado Com Sucesso!

**Por Favor Defina Suas Preferências**

Senha	<input type="text"/>
Gênero	Aventura <input type="button" value="v"/>
Distribuidora	BRASILEIRINHAS <input type="button" value="v"/>
Ator	ASHLEY JUDD <input type="button" value="v"/>
Pais de Produção	ALEMANHA <input type="button" value="v"/>
Idioma	COM LEGENDAS <input type="button" value="v"/>
Confirmar	<input type="button" value="OK"/>

Figura 18 – Tela para definição das preferências primárias do cliente

Após a confirmação das preferências as opções são armazenadas na base de dados, onde serão posteriormente consultados pelo AII a cada visita do cliente.

Com a definição das preferências primárias, na próxima vez que o cliente efetuar o *login* o mesmo será remetido à uma tela com a confirmação do *login* e a recomendação do AII, como é demonstrado na figura 19.

Esta recomendação acontece da seguinte forma, o AII consulta a lista de parâmetros gerados pelo AIC, os parâmetros são o seguinte:

- gênero do título (ação, comédia...),
- língua do título (dublado, legendado...),
- distribuidora do título (Sony pictures...),
- país de produção do título (EUA, Brasil...),
- ator principal do título.

Com estas informações o AII consulta as preferências primárias do cliente, os parâmetros mais locados e os mais consultados pelo cliente.

Após obter a lista final de parâmetros, é gerada uma lista com todos títulos que englobam estes parâmetros, o AII consulta os títulos indicados pelo AIC no dia e indica os títulos ao cliente, armazenando a data da indicação na base de dados para servir de sensor do AII onde, através desta consulta, o agente pode verificar se suas indicações estão surtindo efeito. No apêndice A é apresentado o código fonte dos dois agentes.

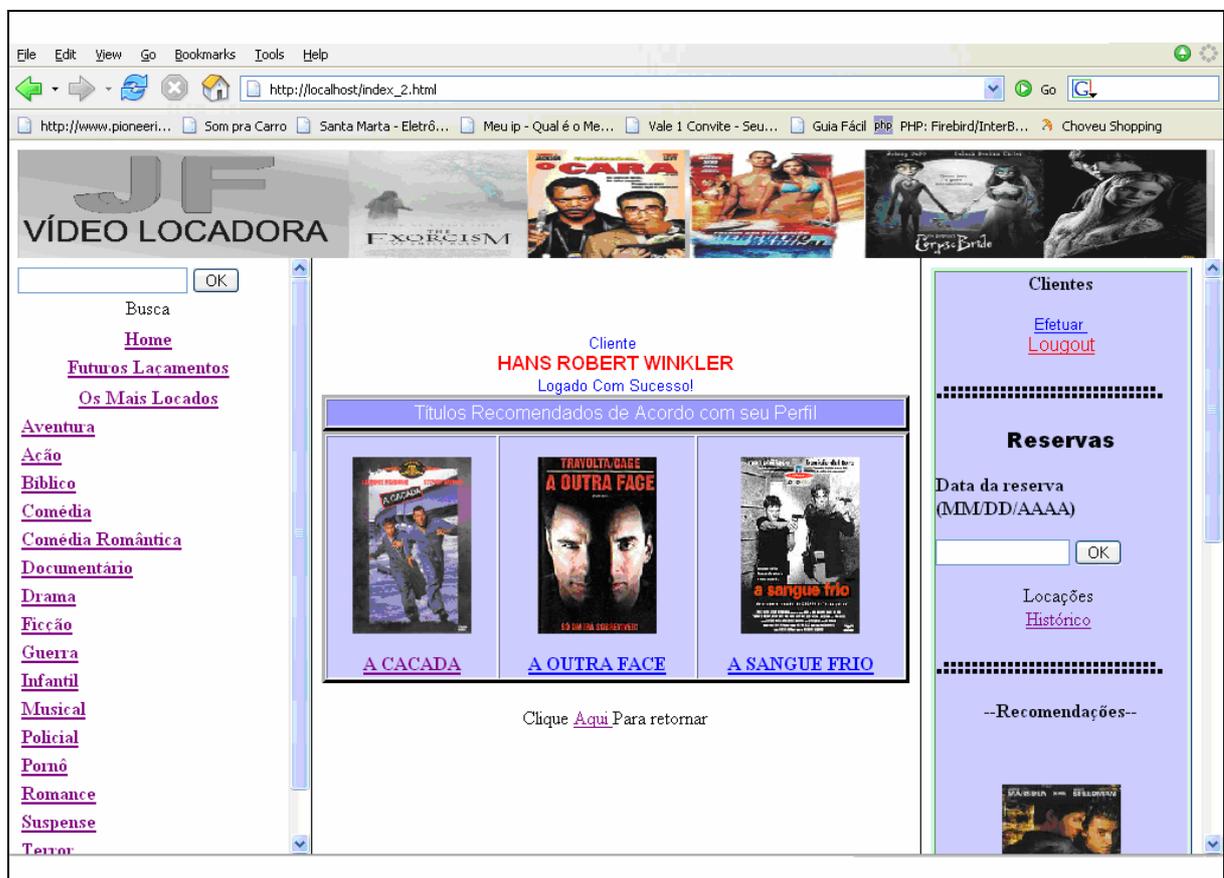


Figura 19 – AII recomenda títulos ao cliente logado

Após o usuário for identificado como cliente, a navegação pode ocorrer normalmente por toda a aplicação, porém no campo de “Busca” da figura 12 agora, o cliente ao realizar uma consulta, será armazenado também o cliente na base de dados, para assim o AIC utilizar desta informação para recomendar com maior perfeição títulos da preferência do cliente.

### 3.6 RESULTADO

A figura 20 apresenta a tela do sistema local, esta tela é responsável pelo cadastro dos clientes. O cliente constante nesta figura é um dos que foram utilizados como testes na aceitação do aplicativo desenvolvido, este cliente utilizou localmente o sistema e através do mesmo fez suas escolhas de títulos, seguindo ou não a recomendação do AII e do AIC.

6.04 - BLACK EYED PEAS - DONT LIE 3:39

**4345**

Dados Do Cliente Últimas Locações Relatórios

3647 Atualização: 9/6/2006 Saldo Devedor R\$ 0,00 Pagar Tudo

Identificação  
Nome: HANS ROBERT WINKLER  
RG: 3R/1842.849 CPF: 460.996.629-87 Nascimento: 31/10/1961

Contato  
Telefone Residencial: Telefone Comercial: Celular:  
E-Mail:  
Trabalho: TORNEARIA IDEAL

Dados Residenciais  
Endereço: ARTHUR KHUL N:177  
Cidade: BLUMENAU Bairro:  
Proximidades: FRENTE LAGOAO LANCHES

Autorizados  
ROBERT

On-Line  
Senha  
Login  
Senha

Inserir  
Alterar  
Deletar  
Consultar  
Sair

WINAMP 05:08

Figura 20 – Cadastro do cliente do programa local

Durante os testes, o AII recomendou filmes de acordo com o perfil do cliente, com isto ele continuou retirando o que lhe era conveniente, neste caso, as preferências primárias também contribuíram, pois, não fugiram no padrão de locações.

Pode-se concluir que o resultado seria diferente se houvesse algum desvio extremo nos parâmetros comuns do cliente, porém as recomendações seriam seriamente diferenciadas havendo desvio nas indicações primárias do cliente, pois o AII, verificaria se o padrão dos parâmetros está de acordo com o que o cliente costuma corriqueiramente locar, ou, posteriormente a implantação do sistema, consultar.

## 4 CONCLUSÕES

Ficou compreendido com este trabalho que o cliente tem sua preferência e seu perfil, e podendo auxiliá-lo em saber o que realmente procura no momento em que procura é benéfico à organização, principalmente quando há a personalização e a individualização por parte da empresa aos seus clientes.

A técnica de agentes de *interface* mostrou-se apta a resolver os problemas levantados neste trabalho. O PHP serviu perfeitamente para o desenvolvimento da aplicação. Os clientes que utilizaram do sistema nas duas semanas de testes mostraram-se satisfeitos com os resultados, e com isso, comprovam que as técnicas aplicadas neste trabalho atingiram seu objetivo.

Apesar de a aplicação ter sido projetada para ser on-line, conseguiu-se apenas rodá-la localmente, com uma aplicação on-line os resultados obtidos com o uso dos agentes de *interface* seriam muito mais satisfatórios.

### 4.2 EXTENSÕES

Como foi projetada para se tornar on-line, porém o seu funcionamento foi todo local, recomenda-se aplicá-lo em um servidor WEB onde o contato com mais usuários o torna mais útil, sendo assim transformada em um possível portal.

Há várias outras formas de aplicar o CRM em uma aplicação, pode-se utilizar, assim como vários *sites* na *Internet*, enviar malas-diretas com motivos comemorativos de aniversários, assim como de indicações de lançamentos de filmes do perfil do cliente, assim como aviso de reserva pelos agentes, onde o próprio agente poderia, ao perceber que há um título novo com o perfil de um cliente assíduo, reservá-lo automaticamente para uma data da

semana que o cliente costuma corriqueiramente locar e avisá-lo que o filme estará disponível.

Portanto, por motivos óbvios de tempo escasso, este trabalho focou no desenvolvimento dos agentes, porém podem ser mais explorado, em trabalhos futuros, os potenciais do CRM na aplicação e a transformação de uma aplicação local para um portal.

## 5 REFERÊNCIAS BIBLIOGRÁFICAS

- ARANHA M. Lúcia; MARTINS M. Helena. **Temas de Filosofia**. São Paulo: Moderna. 1998.
- BARBALHO, Célia R.S. **Portais eletrônicos: estudo comparativo da oferta em comunicação**. [2004]. Disponível em <<http://www.portcom.intercom.org.br/projetos/endocom/endocom2004/Barbalho.PDF>> Acesso em 16 Jun 2006.
- BIGUS J.; BIGUS J.R. **Constructing intelligent agents with Java**. Toronto: Wiley.1997.
- BOGO, Luís H. **Agentes inteligentes para o processo de negociação em um ambiente de comércio eletrônico**. 2000. 90 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau. Blumenau.
- BOTTONI, F.; ZOCCHIO M. Como ter o cliente nas mãos. **Info canal**. Edição 1 2006. Disponível em <[http://info.abril.com.br/canal/edicoes/1/conteudo\\_133600.shtml](http://info.abril.com.br/canal/edicoes/1/conteudo_133600.shtml)> Acesso em 16 Jun 2006.
- BRADSHAW, Jeffrey M. **Software Agents**. Massachusetts: The MIT Press, 1997.
- CABRAL, A. **CRM, a cultura de pensar no cliente**. [S.l.]. Disponível em <[http://www.pauloangelim.com.br/artigos3\\_22.html](http://www.pauloangelim.com.br/artigos3_22.html)> Acesso em: 10 Set 2005.
- CARDOSO, M. S.; GONÇALVES FILHO, C. **CRM em ambiente e-business: como se relacionar com clientes aplicando novos recursos da web**. São Paulo: Atlas, 2001.
- CARVALHO, G; TAVARES M. **Informação e conhecimento: uma abordagem organizacional**. Rio de Janeiro: Quality Mark. 2001.
- FARACO, Rafael A. **Uma arquitetura de agentes para negociação dentro do domínio do comércio eletrônico**. 1998. Disponível em <<http://www.eps.ufsc.br/disserta98/faraco/>> Acesso em 16 Jun 2006.
- FELIPINI, D. **CRM: gerando benefícios ao cliente**. [S.l.], 2002. Disponível em: <<http://www.e-commerce.org.br/Artigos/CRM.htm>> Acesso em: 10 Set 2005.
- FRAIZ, Marcello R. T. **Protótipo de um sistema de comércio eletrônico para livraria virtual utilizando agentes inteligentes**. 2000. 64 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- HANSON, Dennis. **CRM: desvendando o CRM**. [2003]. Disponível em <<http://www.mktdireto.com.br/crm2.html>> Acesso em 16 Jun 2006.
- HUHNS, M.; SINGH, M. **Readins in Agents**. California: MK, 1998.
- JENNIGS, N.; WOOLDRIDGE, M. **Agent Technology**. New York: Springer, 1998.

LIAUTAUD, Bernard. **Inteligência em e-business**: Rio de Janeiro: Qualitymark, 2002.

LOZANO, Fernando. **Programação orientada a objetos com PHP**. [2002]. Disponível em <<http://www.lozano.eti.br/palestras/oo-php.pdf>> Acesso em 16 Jun 2006.

NONAKA K; TAKEUCHI H. **Criação de conhecimento na empresa**. Rio de Janeiro: Campus. 1997.

RIOS, J. A. GED Como ferramenta na gerência do conhecimento explícito organizacional. In: ENCONTRO NACIONAL DE CIÊNCIA DA INFORMAÇÃO, 6., 2005, Bahia. **Anais...** Bahia, 2005. Disponível em: <[http://www.cinform.ufba.br/vi\\_anais/docs/JocelmaRiosGED.pdf](http://www.cinform.ufba.br/vi_anais/docs/JocelmaRiosGED.pdf)>. Acesso em: 10 Set 2005.

RUSSEL, J; NORVIG R. **Artificial Intelligence**: a modern approach. New Jersey: Prentice-Hall International. 1995.

SOUZA, Michel. **CRM**. 2004. Disponível em <<http://www.prodepa.psi.br/sqp/pdf/imasters-crm.pdf>> Acesso em 16 Jun 2006.

STEILEIN, Maria Ap. **[Entrevista realizada em 13 de setembro de 2005, na JF vídeo locadora e games ltda me]**. 2005.

SWIFT, Ronald. **CRM: O revolucionário marketing de relacionamento com o cliente**. Rio de Janeiro : Campus, 2001.

WEITZEN H. Skip. **O poder da informação**: como transformar a informação em um negócio lucrativo. São Paulo: Makron. 1991.

## 6 APENDICE

```

class AIC {
//*****
// Atributos
//*****

    // array que receberá a lista
    // dos filmes indicados
    protected $lista;
    // parâmetros dos títulos mais consultados e mais locados
    protected $parametros;
    // atributo que receberá todas as queries de consulta e de inclusão
    protected $sql;

//*****
// Métodos
//*****

// Obtém parâmetros dos filmes mais locados,
    public function monta_sql_parametros(){
        $this->sql = 'select count(*),
                    f.cd_genero,
                    f.cd_ator_principal,
                    f.cd_dist,
                    f.cd_lingua,
                    f.cd_pais_producao
                    from
                    locacao_filme l,
                    filme f
                    where
                    f.cd_filme = l.cd_filme
                    group by
                    f.cd_genero,
                    f.cd_ator_principal,
                    f.cd_dist,
                    f.cd_lingua,
                    f.cd_pais_producao
                    union
                    select
                    count(*),
                    f.cd_genero,
                    f.cd_ator_principal,
                    f.cd_dist,
                    f.cd_lingua,
                    f.cd_pais_producao
                    from
                    log_consulta c,
                    filme f
                    where
                    f.cd_filme = c.cd_filme
                    group by
                    f.cd_genero,
                    f.cd_ator_principal,
                    f.cd_dist,
                    f.cd_lingua,
                    f.cd_pais_producao
                    order by 1 desc, 2';
    }
}

```

```

    return $this->sql;
}

// É necessário passar o resultado da conexão
public function post_query_parametros($resp){
    $this->parametros = ibase_fetch_row($resp);
}

public function get_parametros(){
    // retorna array com os dados
    return $this->parametros;
}

// Monta query de consulta dos filmes que se encaixam do perfil
// (parâmetros) passado
public function monta_sql_filmes($gen, $ator, $dist, $lingua, $pais){
    $this->sql = 'select
        cd_filme
    from
        filme
    where
        cd_genero = '.$gen.'and
        cd_ator_principal IN ( '.$ator.' )and
        cd_dist = '.$dist.'and
        cd_lingua = '.$lingua.'and
        Cd_pais_producao = '.$pais.' and
        ds_filme not in (select
            f.ds_filme
        from
            indicacao_filme i,
            filme f
        where
            f.cd_filme = i.cd_filme)';

    return $this->sql;
}

// É necessário passar o resultado da conexão
public function post_query_filmes($resp){
    $this->lista = ibase_fetch_row($resp);
}

// retorna array com os dados
public function get_lista(){
    return $this->lista;
}

//
// Verifica se já não há indicacao no dia
// Pois será apenas uma indicacao por dia independente de acessos
public function set_sql_indicacao($data){
    $this->sql = "select
        1
    from
        indicacao
    where
        dt_indicacao = '". $data."'";

    return $this->sql;
}

// Se for = 1 então já houve indicação hoje
public function get_indicacao($ok){

```

```

    if ($ok == 1)
        return true;
    else
        return false;
}

// _____
// popula a tabela com os dados da indicacao
//***** cd_id agente *****
//== 1 para AIC
//== 2 para AII
//*****
public function insere_indicacao($data, $hr){
    $this->sql = "insert into indicacao
                (dt_indicacao,
                 hr_indicacao,
                 cd_id_agente)
                values
                ('. $data .',
                 '. $hr .',
                 1)";
    return $this->sql;
}

// Popula a tabela Indicação_filme com o filme indicado
public function insere_indicacao_filme($data, $hr, $filme){
    $this->sql = "insert into indicacao_filme
                (dt_indicacao,
                 hr_indicacao,
                 cd_filme,
                 cd_id_agente)
                values
                ('.$data.',
                 '$hr.',
                 ". $filme.",
                 1)";
    return $this->sql;
}

// Método que Consulta se o título foi indicado no dia
// Para não haver repetição
public function set_sql_indicados($data){
    $this->sql = "select
                cd_filme
                from
                indicacao_filme
                where
                dt_indicacao = '. $data.'";
    return $this->sql;
}
// _____
} // Fim da Classe AIC

class AII{
//*****
// Atributos
//*****

// Código do Cliente
protected $cliente;

```

```

// Lista gerada pelo AIC
protected $lista_AIC;

// Lista de filmes locados
protected $lista_loc;

// Lista Com os Parâmetros da preferência Primária
protected $lista_pref_primaria;

// Lista com os Parâmetros das locações e consultas do cliente
protected $lista_parametros;

protected $lista;

//*****
// Métodos
//*****
//_____CLIENTE_____
// Seta o cliente
public function set_cliente($cli){
    $this->cliente = $cli;
}

// retorna o cliente
public function get_cliente(){
    return $this->cliente;
}

//_____LISTA_AIC_____
// Obtém a lista dos parametros do AIC
public function set_lista_aic($param_AIC){
    $this->lista_AIC = $param_AIC;
}

// Retorna a lista do AIC
public function get_lista_aic(){
    return $this->lista_AIC;
}

//_____LISTA_FILMES_LOCADOS_____
// Obtém a Lista de filmes locados
public function set_lista_loc($lista){
    $this->lista_loc = $lista;
}

// Retorna a lista de filmes locados
public function get_lista_loc(){
    return $this->lista_loc;
}

//_____PARAMETROS_PREF_PRIMARIA_____
// Query que busca as preferências primárias do cliente
public function set_sql_preferencia_primaria($cli){
    $this->sql = 'select
                cd_genero,
                cd_ator,
                cd_distribuidora,
                cd_lingua,
                cd_pais_producao
            from
            preferencia_primaria
    
```

```

        where
            cd_cliente = '.$cli.';
    return $this->sql;
}

// É necessário passar o resultado da conexão
public function set_preferencia_primaria($resp){
    $this->lista_pref_primaria = ibase_fetch_row($resp);
}

// Retorna a lista dos parâmetros da preferências primárias
public function get_preferencia_primaria(){
    return $this->lista_pref_primaria;
}

// -----
// Verifica as Listas Obtidas Pelo Cliente
// e gera uma nova lista com os valores passados pelas listas
public function compara_lista($lista_param, $lista_pref_prim,
$lista_aic){
    $i = 1;
    while ($lista_param[$i] <> '') {
        if ($lista_param[$i] <> $lista_pref_prim[($i-1)]){
            // se os parametros de locações e consultas for igual ao indicado
            // na lista de preferencia primária
            // verifica se o parametro do aic é igual ao da preferencia
primária
            if ($lista_aic[$i] == $lista_pref_prim[($i-1)])
                // parametros recebe o parametro da lista do aic que é o mesmo
da
                // preferencia primária
                $this->lista_parametros[$i] = $lista_aic[$i];
        }
        $i++;
    }
}

// -----
// Obtêm parâmetros dos filmes mais locados e consultados pelo cliente
public function monta_sql_parametros($cli){
    $this->sql = 'select
                    count(*),
                    f.cd_genero,
                    f.cd_ator_principal,
                    f.cd_dist,
                    f.cd_lingua,
                    f.cd_pais_producao
                from
                    locacao_filme lf,
                    locacao l,
                    filme f
                where
                    f.cd_filme = lf.cd_filme and
                    l.cd_locacao = lf.cd_locacao and
                    l.cd_cliente = '.$cli .'
                group by
                    f.cd_genero,
                    f.cd_ator_principal,
                    f.cd_dist,
                    f.cd_lingua,
                    f.cd_pais_producao
                union

```

```

select
    count(*),
    f.cd_genero,
    f.cd_ator_principal,
    f.cd_dist,
    f.cd_lingua,
    f.cd_pais_producao
from
    consulta_cliente c,
    filme f
where
    f.cd_filme = c.cd_filme and
    c.cd_cliente = '. $cli .'
group by
    f.cd_genero,
    f.cd_ator_principal,
    f.cd_dist,
    f.cd_lingua,
    f.cd_pais_producao
order by 1 desc, 2';

return $this->sql;
}

// É necessário passar o resultado da conexão
public function set_parametros($resp){
    $this->lista_parametros = ibase_fetch_row($resp);
}

// Retorna a lista dos parâmetros da preferências primárias
public function get_parametros(){
    return $this->lista_parametros;
}

// _____LISTA_INDICACAO_____
//*****
// Monta sql para um array com os filmes que se encaixam nos parâmetros de
filmes do cliente
// Não pode constar filmes que ele já locou,
// Nem filmes que já foram indicados
public function monta_sql_filmes($gen, $ator, $dist, $lingua, $pais,
$cli, $lista){
    $this->sql = 'select
        cd_filme
    from
        filme
    where
        cd_genero = '.$gen.'and
        cd_ator_principal IN ( '.$ator.' )and
        cd_dist = '.$dist.'and
        cd_lingua = '.$lingua.'and
        cd_pais_producao = '.$pais.' and
        ds_filme not in (select
            f.ds_filme
        from
            locacao l,
            locacao_filme lf,
            filme f
        where
            l.cd_cliente = '. $cli.' and
            f.cd_filme = lf.cd_filme and

```

```

        lf.cd_locacao = l.cd_locacao) and
        ds_filme not in (select
                        f.ds_filme
                        from
                            indicacao_filme i,
                            filme f
                        where
                            f.cd_filme =

i.cd_filme and
('.$lista[0].',
'.$lista[1].',
'.$lista[2].'))
                        order by dt_lancamento;';

        return $this->sql;
    }
    // É necessário passar o resultado da conexão
    public function post_query_filmes($resp){
        $this->lista = ibase_fetch_row($resp);
    }

    // retorna array com os dados
    public function get_lista(){
        return $this->lista;
    }

    // _____ VERIFICA_INDICACAO _____
    // verifica se o cliente locou recentemente alguma indicação
    // Primeiro verifica se há alguma indicação a um mês

    public function set_sql_rec_mes($cli, $data){
        $this->sql = "select
                    1
                    from
                        indicacao_filme_AII
                    where
                        cd_cliente = ".$cli." and
                        dt_indicacao = ('".$data."'); ";
        return $this->sql;
    }

    public function get_rec_mes($ok){
        // Se == 1 então há indicação a um mês
        if ($ok == 1)
            return true;
        else
            // senão não há recomendação a um mês
            return false;
    }

    public function set_sql_ver_locou($data, $cli, $data){
        $this->sql = "select
                    1
                    from
                        locacao_filme lf,
                        locacao l
                    where
                        l.cd_cliente = ".$cli." and,

```

```

        l.cd_locacao = lf.cd_locacao and
        lf.cd_filme in (select
            f.cd_filme
        from
            indicacao_filme_AII ai,
            reserva_filme r,
            filme f
        where
            ai.cd_cliente = ". $cli." and
            r.cd_cliente = ai.cd_cliente and
            f.cd_filme = r.cd_filme and
            f.cd_filme = ai.cd_filme and
            ai.dt_indicacao between '". $data .' and '".
$data .'') and
        lf.dt_locacao between '". $uma_mes_atras .' and '". $data .
";";
    return $this->sql;
}

// ***** Função Reagente do Agente *****
// Verifica se obteve sucesso nas suas indicações
public function get_ver_locou($ok){
    // Se o cliente locou recentemente
    if ($ok == 1)
        return true;
    else
        // retorna false para não locado os filmes indicados em um mês
        return false;
}

// se o cliente não locou recentemente nenhum filme indicado pelo agente
// o agente utiliza resultado com os parâmetros obtidos
// pela query de consulta de perfil
// com o outros parâmetros
// ----- Deve passar a conexao -----
Public function set_novos_parametros($sql){
    // deve procurar os parâmetros de preferência de segunda opção
(segunda linha da query)
    ibase_fetch_row($sql);
    // armazena o resultado
    // com os parâmetros preferenciais
    $this->lista_parametros = ibase_fetch_row($sql);
}
//


---


// popula a tabela com os dados da indicacao
//***** cd_id agente *****
//== 1 para AIC
//== 2 para AII
//*****
public function insere_indicacao($data, $hr, $cliente){
    $this->sql = "insert into indicacao
        (dt_indicacao,
        hr_indicacao,
        cd_cliente,
        cd_id_agente)
        values
        ('. $data .',
        '. $hr .',
        '. $cliente.',
        2)";
    return $this->sql;
}

```

```
}  
//  
// Popula a tabela Indicação_filme com o filme indicado  
public function insere_indicacao_filme($data, $hr, $filme, $cliente){  
    $this->sql = "insert into indicacao_filme_AII  
        (dt_indicacao,  
         hr_indicacao,  
         cd_filme,  
         cd_id_agente,  
         cd_cliente)  
        values  
        ('".$data."',  
         ".$hr."',  
         ".$filme."',  
         2,  
         ".$cliente.");";  
    return $this->sql;  
}  
  
}  
// FIM DA Classe AII
```