

**UNIVERSIDADE FEDERAL DE PERNAMBUCO**

**CENTRO DE INFORMÁTICA**

**Sistemas de Gerenciamento de Banco de Dados**

*Firebird*

André Ricardo Rolim dos Reis (arrr)

Marina Alecrim de Carvalho Novaes (macn)

Paulo Orlando Vieira de Queiroz Sousa (povqs)

Thaís Melise Lopes Pina (tmlp)

Recife, 2011

**Sumário**

[1. Introdução 2](#_Toc297157747)

[2. Requisitos do SW 4](#_Toc297157748)

[2.1. Convenções para Identificação dos Requisitos 4](#_Toc297157749)

[2.2. Requisitos Funcionais 4](#_Toc297157750)

[2.3. Requisitos Não Funcionais 6](#_Toc297157751)

[3. Diagrama de Casos de Uso e Detalhamentos 8](#_Toc297157752)

[3.1. Convenções para Identificação dos Casos de Uso 8](#_Toc297157753)

[3.2. Estrutura dos casos de uso 8](#_Toc297157754)

[3.3. Prioridades dos casos de uso 8](#_Toc297157755)

[3.4. Diagrama de Caso de Uso 9](#_Toc297157756)

[3.5. Diagramas de Sequência 10](#_Toc297157757)

[3.6. Detalhamento dos Casos de Uso 19](#_Toc297157758)

[4. Arquitetura do Sistema 28](#_Toc297157759)

[5. Banco de Dados 29](#_Toc297157760)

[5.1. Modelo Conceitual 29](#_Toc297157761)

[5.2. Modelo Lógico 29](#_Toc297157762)

[6. Sistema de Bloqueio Explícito no Firebird 30](#_Toc297157763)

[7. Parâmetros de Transação 31](#_Toc297157764)

[7.1. Classificando os parâmetros 31](#_Toc297157765)

[8. Interface Gráfica 32](#_Toc297157766)

[9. Listagem Código Fonte (código) 34](#_Toc297157767)

[9.1. Conexão 34](#_Toc297157768)

[9.2. WITH LOCK 34](#_Toc297157769)

[10. Planejamento da Apresentação 35](#_Toc297157770)

# Introdução

O objetivo principal do desenvolvimento desse projeto é investigar o sistema de bloqueio explícito do banco de dados *Firebird*. O bloqueio é feito através da cláusula WITH LOCK, e os parâmetros de transação que afetam o controle o controle de concorrência realizado por este tipo de bloqueio.

A aplicação desenvolvida para o estudo tem como principal objetivo controlar a locação de filmes numa locadora, a qual dispõe de filmes que podem ser locados por clientes. A investigação do sistema de bloqueio explícito será testada através de diversas chamadas concorrentes.

A aplicação será robusta e aplicará o sistema de bloqueio explicito a fim de evitar que as informações armazenadas na base de dados se tornem inconsistentes após o fim de diversas transações que utilizam os mesmos dados, monitorando as transações mais comuns que existem no sistema de uma locadora: inserir, remover, atualizar e listar clientes, filmes e locações.

# Requisitos do SW

## Convenções para Identificação dos Requisitos

Os requisitos, por convenção, são indicados e referenciados através de uma sigla seguida de um número identificador. Dessa maneira, fica estabelecido que os requisitos funcionais sejam representados pelo formato [RF xx], e os requisitos não funcionais, por sua vez, possuem a forma [RNF xx], onde xx se refere ao número do requisito.

## Requisitos Funcionais

Os requisitos funcionais são aqueles que descrevem o comportamento do sistema, suas ações para cada entrada, ou seja, é aquilo que descreve o que tem que ser feito pelo sistema.

**[RF 01] Inserir cliente**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 01] Inserir cliente | | |
| Casos de Uso relacionados: | [UC 01] | | |
| Descrição: | Permite que ao usuário que insira clientes no sistema. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 02] Remover cliente**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 02] Remover cliente | | |
| Casos de Uso relacionados: | [UC 02] | | |
| Descrição: | Permite que ao usuário que remova clientes existentes no banco de dados do sistema. Uma vez realizada, a operação não pode ser revertida. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 03] Atualizar cliente**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 03] Atualizar cliente | | |
| Casos de Uso relacionados: | [UC 03] | | |
| Descrição: | Permite que o usuário atualize os dados de um determinado cliente cadastrado no sistema. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 04] Listar cliente**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 04] Listar cliente | | |
| Casos de Uso relacionados: | [UC 04] | | |
| Descrição: | Permite que ao usuário que liste todos os clientes existentes no banco de dados. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 05] Inserir filme**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 05] Inserir filme | | |
| Casos de Uso relacionados: | [UC 05] | | |
| Descrição: | Permite que ao usuário que insira filme no sistema. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 06] Remover filme**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 06] Remover filme | | |
| Casos de Uso relacionados: | [UC 06] | | |
| Descrição: | Permite que ao usuário que remova filmes existentes no banco de dados do sistema. Uma vez realizada, a operação não pode ser revertida. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 07] Atualizar filme**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 07] Atualizar filme | | |
| Casos de Uso relacionados: | [UC 07] | | |
| Descrição: | Permite que o usuário atualize os dados de um determinado filme cadastrado no sistema. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 08] Listar filme**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 08] Listar filme | | |
| Casos de Uso relacionados: | [UC 08] | | |
| Descrição: | Permite que ao usuário que liste todos os filmes existentes no banco de dados. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 09] Inserir locação**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 09] Inserir locação | | |
| Casos de Uso relacionados: | [UC 09] | | |
| Descrição: | Permite que ao usuário que insira locação no sistema. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 10] Remover locação**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 10] Remover locação | | |
| Casos de Uso relacionados: | [UC 10] | | |
| Descrição: | Permite que ao usuário que remova locações existentes no banco de dados do sistema. Uma vez realizada, a operação não pode ser revertida. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 11] Atualizar locação**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 11] Atualizar locação | | |
| Casos de Uso relacionados: | [UC 11] | | |
| Descrição: | Permite que o usuário atualize os dados de uma determinada locação cadastrada no sistema. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RF 12] Listar locação**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RF 12] Listar locação | | |
| Casos de Uso relacionados: | [UC 12] | | |
| Descrição: | Permite que ao usuário que liste todas as locações existentes no banco de dados. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

## Requisitos Não Funcionais

Os requisitos não funcionais são restrições sobre os serviços ou as funções oferecidas pelo sistema. Muitos dizem respeito ao sistema como um todo, e não a características individuais deste. Isso significa que eles, freqüentemente, são mais importantes que os requisitos funcionais individuais. Os mesmos estão listados abaixo:

**[RNF 01] Utilizar Linguagem C#**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RNF 01] Utilizar Linguagem C# | | |
| Casos de Uso relacionados: | Todos. | | |
| Descrição: | O sistema deve ser desenvolvido utilizando a linguagem C#. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RNF 02] Utilizar o Banco de Dados Firebird**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RNF 02] Utilizar Banco de Dados Firebird | | |
| Casos de Uso relacionados: | Todos. | | |
| Descrição: | Os dados serão persistidos no banco de dados do tipo Firebird | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RNF 03] Uso da cláusula WITH LOCK**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RNF 03] Uso da cláusula WITH LOCK | | |
| Casos de Uso relacionados: | Todos. | | |
| Descrição: | De acordo com a necessidade de cada transação, o sistema deverá fazer uso da cláusula WITH LOCK para garantir da melhor forma, a durabilidade e consistência dos dados armazenados na base de dados. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RNF 04] Mensagem de erro**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RNF 04] Mensagens de erro | | |
| Casos de Uso relacionados: | Todos. | | |
| Descrição: | As mensagens de erro do sistema deverão ser claras e objetivas. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RNF 05] Tempo de resposta**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RNF 05] Tempo de resposta | | |
| Casos de Uso relacionados: | Todos. | | |
| Descrição: | As consultas realizadas ao sistema não podem ultrapassar o limite máximo de tempo de 10 segundos. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RNF 06] Processamento multiusuário**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RNF 06] | | |
| Casos de Uso relacionados: | Todos. | | |
| Descrição: | O sistema deve ser suportar processamento multiusuário, isto é, permitir que diversos usuários (clientes) possam utilizar o sistema simultaneamente. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

**[RNF 07] Integridade**

|  |  |  |  |
| --- | --- | --- | --- |
| Identificação: | [RNF 07] Integridade | | |
| Casos de Uso relacionados: | Todos. | | |
| Descrição: | Os dados armazenados e consultados do banco de dados do sistema devem estar corretos e coesos com os dados fornecidos ao sistema. | | |
| Prioridade: | 🗹 Essencial | 🞎 Importante | 🞎 Desejável |

# Diagrama de Casos de Uso e Detalhamentos

## Convenções para Identificação dos Casos de Uso

Os casos de uso do sistema são indicados pela sigla UC (do inglês, *Use Case*) e um número [UC xx], onde xx se refere ao número do caso de uso.

Os tópicos seguintes apresentarão a estrutura dos casos de uso, quais os tipos de prioridades são encontrados nos casos de uso e a descrição dos atores.

## Estrutura dos casos de uso

Cada caso de uso terá o seguinte formato:

* **Identificação:** A identificação do caso de uso, todos os casos de uso seguem a notação UC xx, onde XX é um número.
* **Nome:** O nome da funcionalidade que o caso de uso descreve.
* **Referências:** a referência para o(s) requisito(s) funcional (is) que o originou.
* **Prioridade:** Prioridade de implementação deste caso de uso.
* **Atores:** Os modelos de usuário e sistemas que utilizarão o caso de uso.
* **Entradas:** Variáveis que serão passadas ao sistema.
* **Pré-condições:** Condições que devem ser satisfeitas antes de o caso de uso ser executado.
* **Fluxo de Eventos:** O passo a passo das ações realizadas para que o caso de uso seja concluído, podendo incluir fluxos de eventos secundários e/ou alternativos e fluxos de erros.
* **Pós-condições:** Condições que devem ser satisfeitas depois de o caso de uso ser finalizado.

## Prioridades dos casos de uso

Os casos de uso são classificados como:

* **Essencial:** É o caso de uso indispensável ao funcionamento do sistema. Esse tipo de caso de uso deve ser implementado impreterivelmente, caso contrário, o projeto perderá sua utilidade.
* **Importante:** Sem este caso de uso, o sistema ainda é capaz de ser utilizado. Contudo, essa utilização se dá de forma não satisfatória pelo cliente.
* **Desejável:** Esse tipo de caso de uso poderá ser implementado em versões posteriores do sistema, visto que, mesmo sem a sua implementação, o sistema atende as suas funcionalidades básicas.

## 

## Diagrama de Caso de Uso

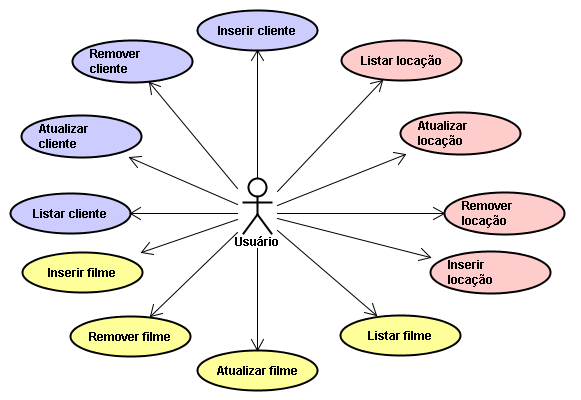


Figure 1. Diagrama de caso de uso

## 

## Diagramas de Sequência

**Inserir Cliente**



**Atualizar Cliente**



**Remover Cliente**



**Inserir Filme**



**Atualizar Filme**



**Remover Filme**



**Inserir Locação**



**Atualizar Locação**



**Remover Locação**



## Detalhamento dos Casos de Uso

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 01 | Inserir cliente | | | |
| ***Referências*** | RF 01 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Dados do cliente: CPF, nome, data de nascimento, endereço e telefone | | | |
| ***Pré-condições*** | | Não se aplica. | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Cliente* e clica em *Inserir*. 2. O sistema exibe um formulário com as informações necessárias para a inserção do cliente. 3. O usuário informa ao sistema os dados requisitados e solicita a inserção, clicando no botão *Inserir Cliente*. 4. O sistema armazena as informações no banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode cancelar a inserção a qualquer momento. 2. O caso de uso é encerrado.   Fluxo Secundário 2:   1. Os dados do cliente fornecidos pelo usuário estão incorretos. 2. O sistema informa ao usuário que os dados informados estão incorretos. 3. Caso de uso retorna ao passo 2 do fluxo principal.   Fluxo secundário 3:   1. O usuário informa dados de um cliente que já existe. 2. Uma mensagem de alerta é exibida para o usuário, impedindo a inserção. 3. O fluxo do caso de uso retorna ao passo 2 do fluxo principal. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O sistema informa ao usuário que algum erro inesperado ocorreu.   3. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | As informações referentes ao novo cliente inserido estão armazenadas no banco de dados. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 02 | Remover cliente | | | |
| ***Referências*** | RF 02 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Não se aplica | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Cliente* e seleciona o cliente a ser removido. 2. Aperta o botão *Remover*. 3. O sistema remove o cliente escolhido do banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode desfazer a seleção antes de confirmar a remoção 2. O caso de uso é encerrado. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | O cliente selecionado pelo usuário será removido do banco de dados | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 03 | Atualizar cliente | | | |
| ***Referências*** | RF 03 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Novos dados do cliente a ser atualizado. | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Cliente* e seleciona o cliente a ser alterado. 2. Altera os valores desejados e clica em *Atualizar Cliente*. 3. O sistema altera o cliente escolhido do banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode cancelar a atualização a qualquer momento. 2. O caso de uso é encerrado.   Fluxo Secundário 2:   1. Os dados do cliente fornecidos pelo usuário estão incorretos. 2. O sistema informa ao usuário que os dados informados estão incorretos. 3. Caso de uso retorna ao passo 2 do fluxo principal.   Fluxo secundário 3:   1. O usuário informa dados de um cliente que já existe. 2. Uma mensagem de alerta é exibida para o usuário, impedindo a atualização. 3. O fluxo do caso de uso retorna ao passo 2 do fluxo principal. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O sistema informa ao usuário que algum erro inesperado ocorreu.   3. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | O banco de dados será atualizado para armazenar as novas informações do cliente. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 04 | Listar cliente | | | |
| ***Referências*** | RF 04 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Não se aplica | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Cliente*. 2. O sistema exibe todos os clientes existentes na base de dados. | | |
| ***Fluxo Secundário*** | | Não se aplica | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | Os dados dos clientes. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 05 | Inserir filme | | | |
| ***Referências*** | RF 05 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Dados do filme: nome, preço, gênero, ano de lançamento e imagem | | | |
| ***Pré-condições*** | | Não se aplica. | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Filme* e clica em *Inserir*. 2. O sistema exibe um formulário com as informações necessárias para a inserção do filme. 3. O usuário informa ao sistema os dados requisitados e solicita a inserção, clicando em *Inserir Filme*. 4. O sistema armazena as informações no banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode cancelar a inserção a qualquer momento. 2. O caso de uso é encerrado.   Fluxo Secundário 2:   1. Os dados do filme fornecidos pelo usuário estão incorretos. 2. O sistema informa ao usuário que os dados informados estão incorretos. 3. Caso de uso retorna ao passo 2 do fluxo principal.   Fluxo secundário 3:   1. O usuário informa dados de um filme que já existe. 2. Uma mensagem de alerta é exibida para o usuário, impedindo a inserção. 3. O fluxo do caso de uso retorna ao passo 2 do fluxo principal. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O sistema informa ao usuário que algum erro inesperado ocorreu.   3. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | As informações referentes ao novo filme inserido estão armazenadas no banco de dados. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UC 06 | Remover filme | | | |
| ***Referências*** | RF 06 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Não se aplica | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Filme* e seleciona o filme a ser removido. 2. Aperta o botão *Remover.* 3. O sistema remove o filme escolhido do banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode desfazer a seleção antes de confirmar a remoção 2. O caso de uso é encerrado. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | O filme selecionado pelo usuário será removido do banco de dados | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 07 | Atualizar filme | | | |
| ***Referências*** | RF 07 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Novos dados do filme a ser atualizado. | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Filme* e seleciona o filme a ser alterado. 2. Altera os valores desejados e clica em *Atualizar Filme*. 3. O sistema altera o filme escolhido do banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode cancelar a atualização a qualquer momento. 2. O caso de uso é encerrado.   Fluxo Secundário 2:   1. Os dados do filme fornecidos pelo usuário estão incorretos. 2. O sistema informa ao usuário que os dados informados estão incorretos. 3. Caso de uso retorna ao passo 2 do fluxo principal.   Fluxo secundário 3:   1. O usuário informa dados de um filme que já existe. 2. Uma mensagem de alerta é exibida para o usuário, impedindo a atualização. 3. O fluxo do caso de uso retorna ao passo 2 do fluxo principal. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O sistema informa ao usuário que algum erro inesperado ocorreu.   3. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | O banco de dados será atualizado para armazenar as novas informações do filme. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 08 | Listar filme | | | |
| ***Referências*** | RF 08 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Não se aplica | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Filme*. 2. O sistema exibe todos os filmes existentes na base de dados. | | |
| ***Fluxo Secundário*** | | Não se aplica | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | Os dados dos filmes. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 09 | Inserir locação | | | |
| ***Referências*** | RF 09 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Dados da locação: Cliente e filmes que serão locados | | | |
| ***Pré-condições*** | | Não se aplica. | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Locação* e seleciona o cliente. 2. O sistema exibe uma listagem dos filmes existentes. 3. O usuário informa ao sistema os filmes que serão locados e solicita a inserção, clicando em *Inserir Locação*. 4. O sistema armazena as informações no banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode cancelar a inserção a qualquer momento. 2. O caso de uso é encerrado. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O sistema informa ao usuário que algum erro inesperado ocorreu.   3. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | As informações referentes à nova locação inserida estão armazenadas no banco de dados. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UC 10 | Remover locação | | | |
| ***Referências*** | RF 10 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Não se aplica | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Locação* e seleciona a locação a ser removida 2. Aperta o botão *Remover*. 3. O sistema remove a locação escolhido do banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode desfazer a seleção antes de confirmar a remoção 2. O caso de uso é encerrado. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | A locação selecionada pelo usuário será removida do banco de dados | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 11 | Atualizar locação | | | |
| ***Referências*** | RF 11 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Novos dados da locação a ser atualizado. | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Locação* e seleciona a locação a ser alterada. 2. Altera o status da locação e clica em *Atualizar*. 3. O sistema altera a locação escolhido do banco de dados. | | |
| ***Fluxo Secundário*** | | Fluxo Secundário 1:   1. O usuário pode cancelar a atualização a qualquer momento. 2. O caso de uso é encerrado.   Fluxo Secundário 2:   1. Os dados da locação fornecidos pelo usuário estão incorretos. 2. O sistema informa ao usuário que os dados informados estão incorretos. 3. Caso de uso retorna ao passo 2 do fluxo principal.   Fluxo secundário 3:   1. O usuário informa dados de uma locação que já existe. 2. Uma mensagem de alerta é exibida para o usuário, impedindo a atualização. 3. O fluxo do caso de uso retorna ao passo 2 do fluxo principal. | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O sistema informa ao usuário que algum erro inesperado ocorreu.   3. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | O banco de dados será atualizado para armazenar as novas informações da locação. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identificação*** | ***Nome*** | | | |
| UC 12 | Listar locação | | | |
| ***Referências*** | RF 12 | | ***Prioridade*** | Essencial |
| ***Atores*** | Usuário | | | |
| ***Entradas*** | Não se aplica | | | |
| ***Pré-condições*** | | Não se aplica | | |
| ***Fluxo Principal*** | | 1. O usuário escolhe a aba *Locação*. 2. O sistema exibe todas as locações existentes na base de dados. 3. Clicando na locação é possível ver detalhes da mesma. | | |
| ***Fluxo Secundário*** | | Não se aplica | | |
| ***Fluxo de erro*** | | * 1. O sistema não consegue se conectar ao banco de dados.   2. O caso de uso é encerrado. | | |
| ***Pós-condições*** | | Os dados das locações. | | |

# Arquitetura do Sistema

A arquitetura do projeto implementado (locadora) é mostrada no diagrama abaixo. A GUI é o componente mais externo na estrutura, sendo através dela que é feita a comunicação com o usuário. As classes pertencentes ao componente Controle possuem as regras de negócio do sistema, as quais fazem o tratamento dos dados. O componente Dados faz a conexão com o banco de dados e, também encapsula os dados do banco.

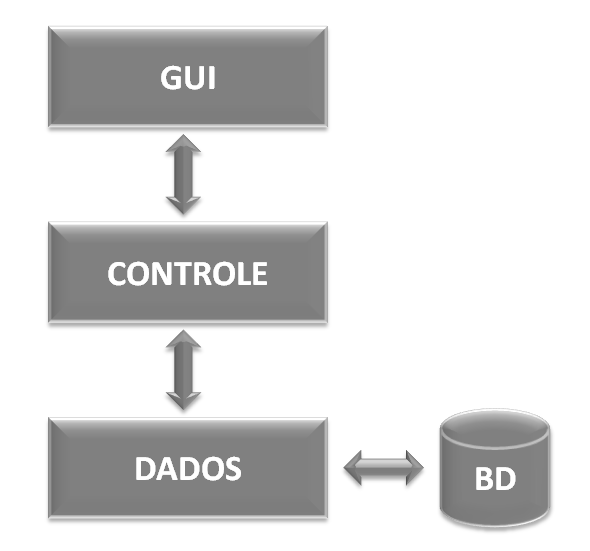


Figure 2. Arquitetura do sistema

# Banco de Dados

## Modelo Conceitual

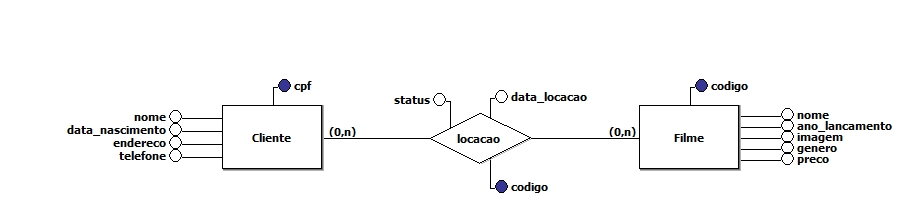


Figure 3. Modelo conceitual

## Modelo Lógico

## logico.jpg

Figure 4. Modelo lógico

# Sistema de Bloqueio Explícito no Firebird

O *Firebird* possui um recurso para o bloqueio explícito de tabelas e transações. Através de comandos SQL adicionados com a cláusula WITH LOCK, podemos bloquear dados. Esse poder adicional de controle de bloqueios permite um melhor gerenciamento dos dados contidos na base de dados, além de um melhor controle no que se diz respeito ao acesso concorrente aos dados armazenados no SGBD. A sintaxe da cláusula WITH LOCK é mostrada na figura 5 e os comandos os quais a cláusula WITH LOCK não está disponível são mostrados na tabela 1.

Quando a cláusula WITH LOCK é executada com sucesso, será estabelecido um bloqueio nas linhas selecionadas que impedirá que outras transações obtenham acessos de escrita a qualquer uma delas, ou de suas dependentes, até o fim da transação. Caso a cláusula FOR UPDATE seja utilizada, o bloqueio será aplicado a todas as linhas, uma a uma. Assim, é possível que um bloqueio que tenha sido bem sucedido ao ser requerido, falhe subseqüentemente, quando for feita uma tentativa de carregar uma linha sobre a qual já tenha sido obtido um bloqueio por outra transação.

Os usuários do banco de dados que implemente o sistema de bloqueio explícito têm a garantia da integridade dos dados, pois o sistema evita que outros usuários modifiquem o conteúdo dos registros envolvidos na transação antes da mesma terminar.

O bloqueio das tabelas e registros é automaticamente desfeito quando a transação é concluída, o que é feito através do comando commit.

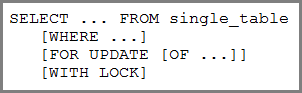


Figure 5. Sintaxe do WITH LOCK

Table 1. Comandos nos quais a cláusula WITH LOCK não está disponível

|  |
| --- |
| Subconsultas |
| Views |
| Consultas que utilizam qualquer tipo de JOIN |
| Consultas que utilizam o operador DISTINCT, a cláusula GROUP BY ou qualquer outra operação de agregação |
| Funções que retornam dados em forma de cursor ou tabela |
| Consultas a tabelas que não pertencem ao SGBD que você está trabalhando, mas sim a outro SGBD (quando se consulta tabela por meio de *linked Server*) |

# Parâmetros de Transação

Uma transação pode ser configurada para trabalhar de diferentes maneiras. Isso pode ser feito através de um conjunto de parâmetros armazenados em uma área da transação chamada de Transaction Parameter Buffer (TPB), pois se pode ter alguns problemas lógicos quando existem várias transações atuando num mesmo banco de dados.

Cada transação possui sua própria TPB e os parâmetros são atribuídos no momento de sua ativação. Existe um conjunto de valores padrão para o caso de não se passar nenhum valor.

## Classificando os parâmetros

* + 1. **Nível de isolamento**

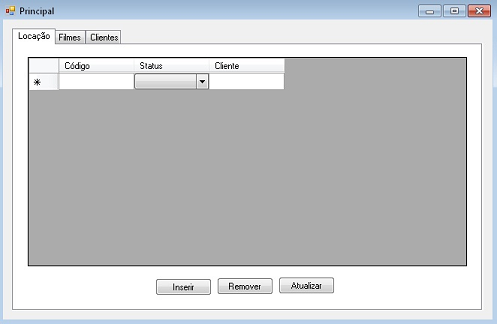
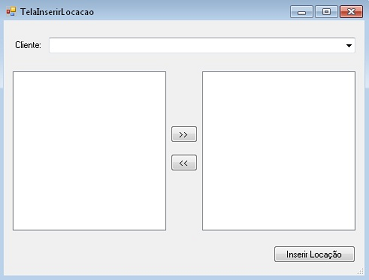
Definição: Especifica se as transações se elas são visíveis ou não para outras transações até o *commit*.

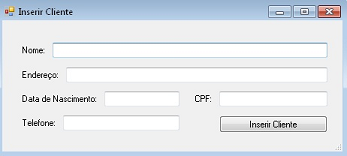
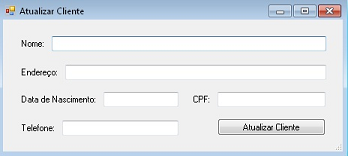
* *Consistency*: Esse nível de isolamento é o mais restrito. As tabelas são travadas para que nenhuma outra transação concorrente possa ter acesso de escrita (gravar). A transação detém o bloqueio estrito da tabela.
* *Concurrency* (default): Permite o acesso concorrente entre transações. Uma transação será capaz de ler informações de uma tabela que estão sendo alteradas por outras transações mesmo que estas não tenham sido *commitadas*. Importante lembrar que uma transação iniciada configurada como concurrency não conseguirá gravar em registros modificados ou *commitados* por outras transações concorrentes a esta.
* *Read\_committed*: Implementa as características de concurrency, porém agora, a transação pode modificar informações que foram *commitadas* por outras transações depois que ela foi iniciada.
  + 1. **Resolução de conflito**

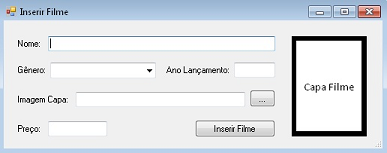
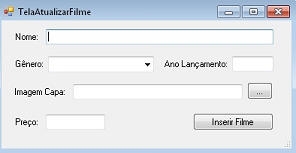
Definição: A ação que será tomada, caso haja um conflito de acesso durante a escrita.

* *Wait* (default): Indica que a transação irá esperar até que uma resolução do conflito seja efetuada e então tentar novamente.
* *Nowait*: Indica que a transação irá retornar uma mensagem de erro assim que ocorrer um bloqueio de uma tabela.

# Interface Gráfica

# Listagem Código Fonte (código)

## Conexão

## WITH LOCK

# Planejamento da Apresentação