**PLANO DE TESTE**

*Apresenta o planejamento para execução do teste, incluindo a abrangência, abordagem, recursos e cronograma das atividades de teste. Identifica os itens e as funcionalidades a serem testados, as tarefas a serem realizadas e os riscos associados com a atividade de teste.*

**a)   Escopo**

Homologação Ferramenta de busca.

**b)   Casos de Teste**Execução Ferramenta de busca, permitindo a localização de operações

a partir de quaisquer propriedades configuradas >

**b.1. Dados**

|  |  |
| --- | --- |
| **Descrição** | Apresentação da Homologação Ferramenta de busca. |
| **Precondições** | Logar no aplicativo Iris, clicar na barra de Ferramenta. |

**b.2. Procedimento**

|  |  |  |
| --- | --- | --- |
| **Procedimento** | **Verificação** | **Resultado** |
| 1. Acessar a área designer. | 1. Efetuar conexão com o banco, gerar um layout e Resulteste. Selecionar o objeto e clicar e na área do Designer em cima do objeto e logo após clicar na opção propriedades:  . | <Resultado observado, como positivo> |
|  |  |  |

### Propriedades dos Resultsets

* Continuar em caso de erro – Indica se a gravação de atualizações no banco deve ou não ser interrompida em caso de erros.
* Registros por lote – Indica a quantidade de registros que será enviada para o banco por lote de transações.
* Descrição – Descrição do resultset. Será exibida na área de descrição detalhada em um painel de propriedades.
* Nome – O nome interno do resultset. Deve ser um nome válido pelos padrões de nomenclatura da linguagem C#.
* RS Externo – Indica se será gerada uma propriedade expondo este resultset em projetos compilados.
* Texto de Apresentação – Nome que será exibido para usuários finais de projetos IRIS compilados, em painéis de propriedades.
* Chave primária – A chave primária da tabela em memória. Pode ser deduzida a partir do schema do banco, ou pode ser informada diretamente no editor.
* Columns – A coleção de colunas do resultset. Deve ser usada apenas para visualização e para testes eventuais, e não deve ser alterada diretamente, pois alterações feitas aqui não serão gravadas. Ao invés disso, devem ser utilizadas operações que alterem as propriedades das colunas, ou mesmo operações que criem novas colunas e relacionamentos.
* Connection – A conexão à qual o resultset está ligado. Não informada para resultsets em memória.
* Registros – Propriedade apenas de leitura informando a quantidade de registros no resultset.
* Table – Exibe as propriedades do objeto DataTable subjacente ao resultset, e permite a inspeção direta dos dados.
* TimeOut - Configura a propriedade CommandTimeOut dos comandos do resultset. Não deve ser confundido com a propriedade ConnectionTimeOut, das strings de conexão.
* Virtual – Indica se o resultset está trabalhando apenas em memória (true) ou se está conectado a um banco de dados (false)
* Errors – Propriedade apenas de leitura, que relata a quantidade de erros encontrados no resultset, quer se trate da violação de constraints em memória, ou de erros de aplicação de atualização para o banco de dados.
* HasError - Propriedade apenas de leitura que indica se o resultset possui ou não erros.
* Relacionamentos Detalhe – Coleção de relacionamentos nos quais o resultset participa como tabela mestre
* Relacionamentos Mestre – Coleção de relacionamentos nos quais o resultset participa como tabela detalhe.
  + Obs.: Da mesma forma que na coleção de colunas, alterações de relacionamentos não devem ser feitas diretamente nestas propriedades, mas sim através de operações específicas para este fim.
* Gerar Delete – Indica se o resultset deve gerar comandos de delete ao aplicar atualizações no banco de dados
* Gerar Insert - Indica se o resultset deve gerar comandos de insert ao aplicar atualizações no banco de dados
* Gerar Update - Indica se o resultset deve gerar comandos de update ao aplicar atualizações no banco de dados
* Select – A sentença SQL utilizada para preencher o resultset
* SQL Delete – Especifica um comando SQL personalizado para exclusão de registros no banco
* SQL Insert - Especifica um comando SQL personalizado para inclusão de registros no banco
* SQL Update - Especifica um comando SQL personalizado para atualização de registros no banco
  + Estas três propriedades só precisam ser preenchidas quando os comandos SQL gerados automaticamente não forem satisfatórios, como no caso de resultsets preenchidos por sentenças com Joins, Unions, etc.

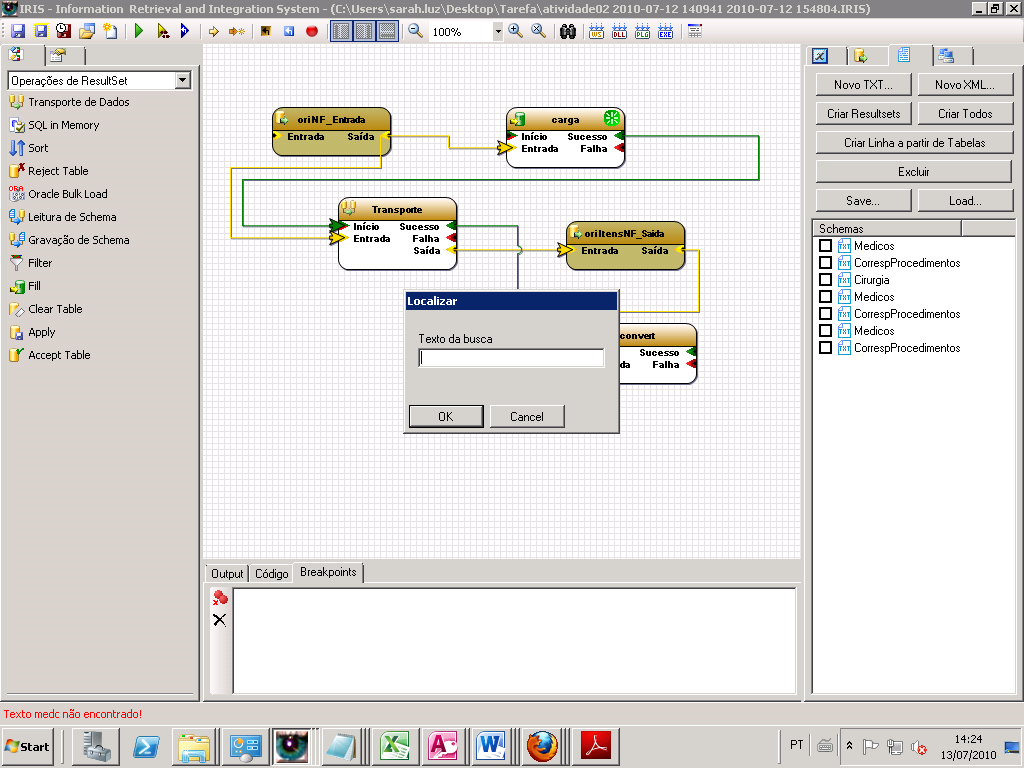
**c)   Observações**

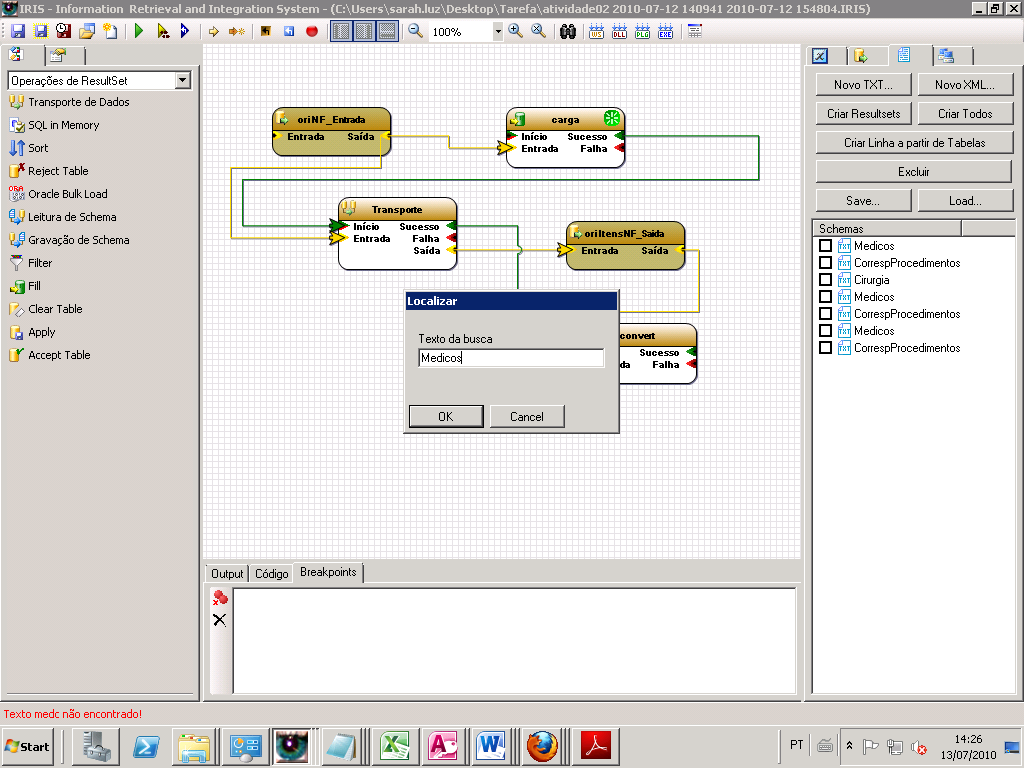
<Realizar teste juntamente com o Marcelo>

**d) Melhorias:**

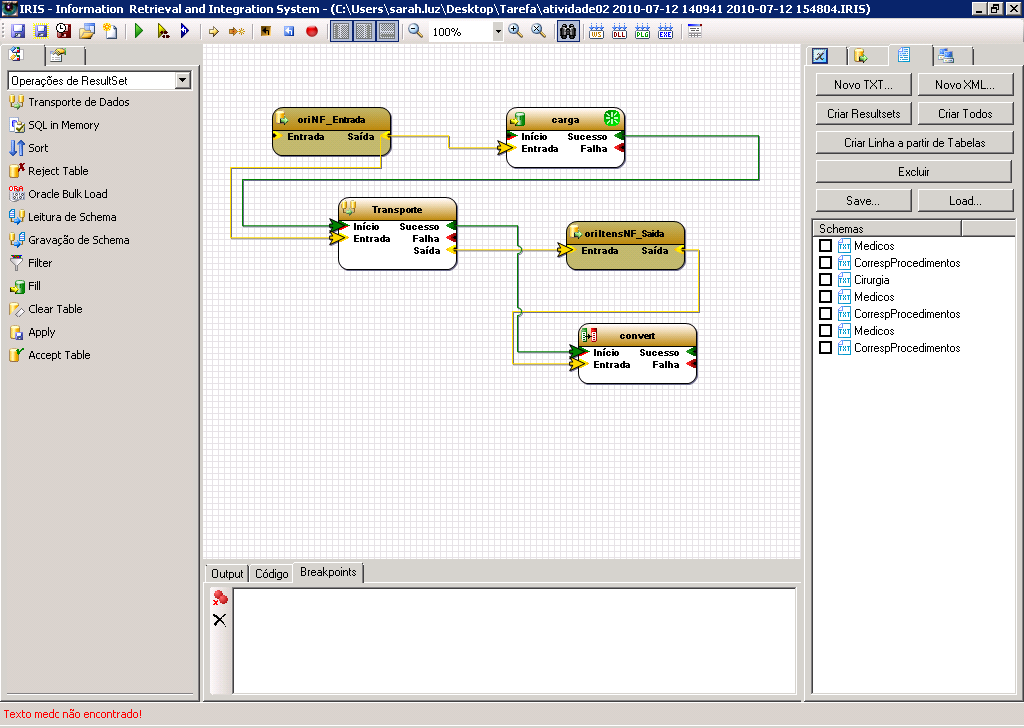
**Evidências de Teste**

*Este documento é o resultado da execução de casos de testes, ou até de etapas de testes inteiras. Ele contém os artefatos gerados nos testes que comprovam que eles foram executados com sucesso, como prints de tela, logs de execução e vídeos.****Testes Realizados: Busca***





Realização do teste: Negativa:



A pesquisa não ler os caracteres a serem pesquisados.