

4 Caso de Uso no Ambiente Oracle

No capítulo anterior foi definido o processo para definição de uma estratégia de rastreabilidade. Neste capítulo será realizada uma instanciação do processo em um ambiente Oracle.

4.1. Instanciação do Processo no Ambiente Oracle

Para aplicar o processo em um caso real, foi utilizado neste trabalho um grande sistema de logística desenvolvido em Oracle PL/SQL. O sistema apresenta mais de 350.000 linhas de código, 4680 elementos entre classes Java e procedures Oracle. O sistema consiste na implementação de mais de 60 casos de uso e 150 regras de negócio.

A seguir é descrita a instanciação e exemplificação de cada etapa do processo levando em conta necessidades específicas do projeto acima descrito e considerando um determinado escopo.

4.2. Instanciação: Definir Estratégia

Foi mencionada pelo gerente do projeto acima citado, a dificuldade encontrada em obter a rastreabilidade entre os requisitos e as procedures Oracle. Esta dificuldade tem uma grande relevância no trabalho dos profissionais envolvidos (por exemplo, Desenvolvedores e Engenheiros de Requisitos), tanto para os profissionais iniciantes quanto para os que já atuam em atividades de manutenção do software. O projeto, em evolução por mais de 10 anos, foi concebido com uma arquitetura onde a grande maioria de regras de negócio foi implementada em procedures Oracle.

Para a instanciação da primeira etapa do processo Definir a Estratégia, devem ser executadas as suas respectivas atividades que podem ser visualizadas na Tabela 9 e que estão organizadas na forma de perguntas da técnica 5W1H.

No entanto, antes de definir a estratégia, utilizou-se a Tabela 5 como ponto de partida. Com base na Tabela 5, onde estão listados alguns exemplos de cenário de uso, foram identificados os seguintes exemplos de cenários como necessidades a serem abordadas:

- a) A navegação entre especificação, projeto, teste e código via rastros (item 4.a da Tabela 5).
- b) Analisar a cobertura dos requisitos em código fonte (item 3.a da Tabela 5).
- c) Análise do impacto da mudança para determinar artefatos afetados por uma extensão de funcionalidade (item 6.a da Tabela 5).

De acordo com os cenários acima, foi possível a identificação dos rastros necessários e suas estratégias.

A seguir foi elaborada uma tabela com a utilização da técnica 5W1H para definição das estratégias.

Tabela 9 – Utilização da técnica 5W1H na definição das estratégias

What	Why	Who	When	How
1- Identificar que rastro é necessário.	2- Justificar porque os rastros são necessários.	3- Identificar os atores de uso da rastreabilidade.	4- Identificar quando os rastros serão utilizados.	5- Identificar a forma de construção e utilização dos rastros.
Rastro entre: a) Especificação de requisitos (UC's) e código (procedures Oracle). b) RN's e código. c) Contexto e código.	Analisar a cobertura dos requisitos em código fonte.	Engenheiro de Requisitos, Gerente de Projeto.	Ao finalizar uma iteração.	Verificação automática com ferramenta de apoio.
	Acompanhamento do estado do requisito.	Gerente de Projeto.	Durante uma iteração.	Verificação automática com ferramenta de apoio.
	Análise do impacto da mudança.	Gerente de Projeto.	Início de uma iteração.	Verificação automática com ferramenta de apoio

4.3. Instanciação: Desenhar

Nesta etapa, são realizadas as atividades de projeto listadas abaixo:

- 1- Elaborar o modelo de uso;
- 2- Identificar os artefatos rastreáveis;
- 3- Identificar elos de rastreabilidade;
- 4- Desenvolver uma arquitetura de rastreabilidade.

4.3.1. Primeira atividade: Elaborar o modelo de uso.

O modelo de uso foi construído de acordo com a descrição detalhada desta etapa do processo.

Foram elaboradas figuras ovais que representam os atores dos cenários de uso selecionados. Dentro das figuras são representadas as metas de uso de cada ator (Figura 17).

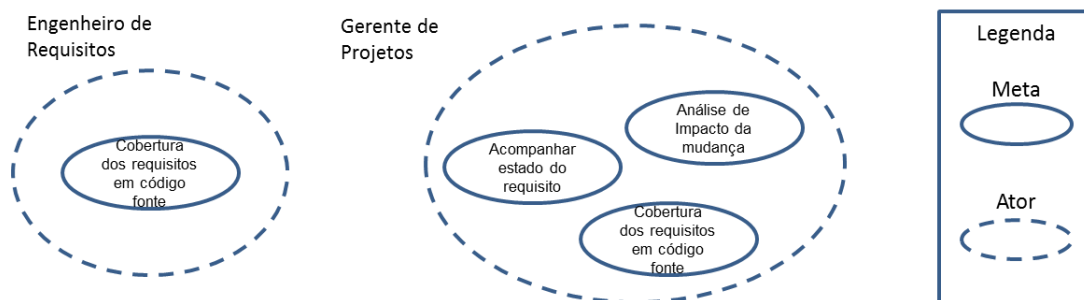


Figura 17 – Modelo de uso para o processo instanciado

4.3.2. Segunda atividade: Identificar os artefatos rastreáveis.

Os artefatos rastreáveis, identificados neste projeto para a estratégia definida, de acordo com a descrição da seção 3.2.2 são:

- ✓ Os documentos de especificação funcional (Casos de Uso);
- ✓ Regras de Negócio;
- ✓ Contexto (Divisão funcional ou modular de um sistema);
- ✓ O código fonte.

4.3.3. Terceira atividade: Identificar os elos de rastreabilidade.

Após verificar os artefatos rastreáveis do projeto, os seguintes elos de rastreabilidade foram identificados, de acordo com as metas definidas e a descrição da seção 3.2.2:

- Casos de Uso _____ Código;
- Regras de Negócio ____ Código;
- Contexto _____ Código.

4.3.4. Quarta atividade: Desenvolver uma arquitetura de rastreabilidade.

A arquitetura apresenta informações de rastreabilidade coletadas entre os artefatos do projeto e suas relações.

A arquitetura apresentada na Figura 18 foi construída identificando os artefatos do projeto rastreáveis e os rastros necessários, de acordo com as seções 4.3.2 e 4.3.3.

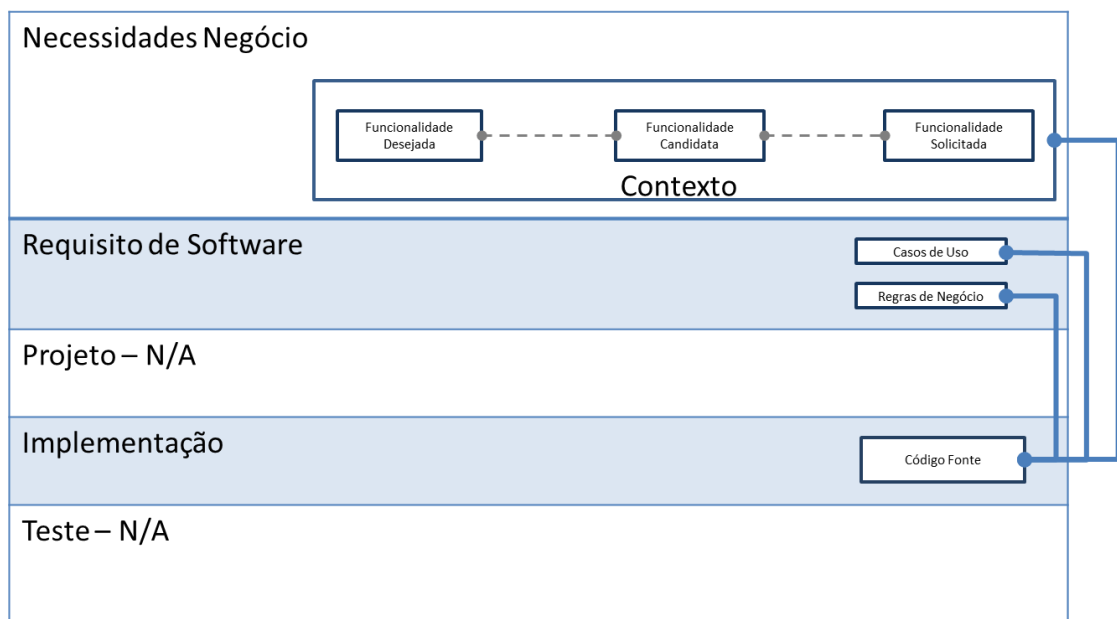


Figura 18 – Arquitetura de rastreabilidade para o processo instanciado.

Ao completar as tarefas desta atividade, é possível verificar qual estratégia de rastreabilidade será utilizada no projeto de desenvolvimento e se a estratégia é explicitamente definida. Além disso, verificam-se quais rastros foram capturados e entre quais artefatos, quem utiliza os rastros, como os rastros são utilizados e por que os rastros são utilizados.

4.4.

Instanciação: Implementar

As seguintes atividades compõem a etapa Implementar:

- 1- Priorizar usos;
- 2- Definir a forma de implementação;
- 3- Escolher a ferramenta;
- 4- Implementar;
- 5- Testar.

4.4.1. Primeira atividade: Priorizar usos.

De acordo com as estratégias definidas na primeira etapa do processo, todos os cenários de uso selecionados serão utilizados na instanciação.

4.4.2. Segunda atividade: Definir a forma de implementação.

Foi definido pelo projeto que a implementação da rastreabilidade deve ser construída de forma semiautomática o que implica na construção de uma ferramenta de apoio.

4.4.3. Terceira atividade: Escolher ferramenta ou artefato de apoio.

Conforme descrito na atividade anterior, a construção da rastreabilidade deve ser construída de forma semiautomática, o que implica na construção de uma ferramenta de apoio.

Conforme as necessidades de rastreabilidade identificadas nas atividades anteriores, a ferramenta deve permitir principalmente a rastreabilidade entre requisitos e código num ambiente Oracle PL/SQL. Também deve permitir a conexão ao SGBD Oracle e validar um padrão de instrumentação. A

instrumentação do código é realizada previamente de forma manual, nos cabeçalhos de todas as procedures e functions PL/SQL. Desta forma, o ambiente Oracle vai estar preparado para a construção de vários tipos de consultas às procedures (p. ex.: por requisitos, por parâmetros de entrada e saída, por contexto do sistema, etc.), obtendo com isso, principalmente, a rastreabilidade necessária entre representações de casos de uso e código PL/SQL.

A ferramenta também deve oferecer ao usuário a possibilidade de visualizar em um relatório, quais “procedures” e/ou “functions” em PL/SQL, estão em desacordo com o padrão previamente estabelecido.

Na Figura 19 é apresentada uma visão geral da ferramenta. As representações de requisitos devem ser inseridas no sistema ao mesmo tempo em que é mantida a instrumentação do código PL/SQL. Na Figura 19 essas tarefas estão ilustradas, à esquerda da figura. Os usuários podem consultar a rastreabilidade entre o código e as representações de requisitos, assim como validar o padrão de instrumentação do código de acordo com a ilustração à direita da figura.

Mais detalhes da Ferramenta são apresentados na seção 4.7.

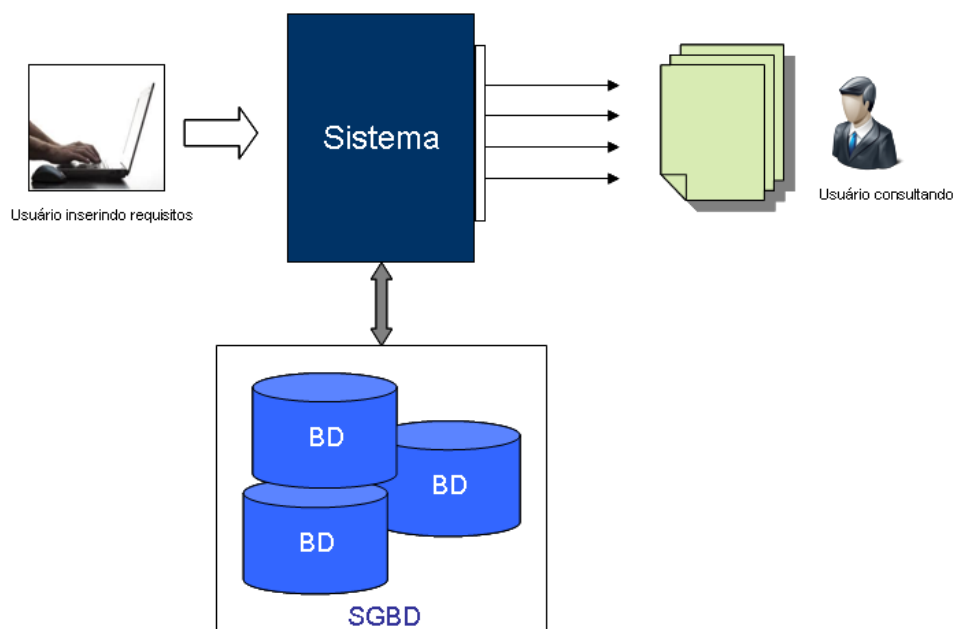


Figura 19 – Visão geral da ferramenta

4.4.4. Quarta atividade: Implementar.

Após a construção da ferramenta de apoio, é necessário criar os elos de rastreabilidade de acordo com os rastros indicados na etapa de definição da estratégia. Desta forma, de posse da ferramenta de apoio, a partir do momento que são identificados os cenários de uso e são definidas as estratégias, os atores responsáveis pelos cenários podem cadastrar as representações de requisitos.

Desta forma, a partir do momento em que são identificados os cenários de uso e são definidas as estratégias, durante as duas atividades anteriores desta etapa do processo, já é possível realizar o cadastro das representações de requisitos pelos atores responsáveis pelos cenários de uso utilizando a ferramenta. Na Figura 20 essa tarefa é realizada conforme indicado pela palavra “Cadastro” (A Figura 20 apresenta uma visão mais detalhada da ferramenta). A instrumentação do código Oracle indicada na Figura 20 deve ser realizada manualmente quando da manutenção (corretivas e evolutivas) das procedures e de acordo com os itens cadastrados na ferramenta. As representações de requisitos cadastrados na ferramenta podem simbolizar, por exemplo, e conforme os cenários desta instanciação, os casos de uso.

A ferramenta quando iniciada, realiza uma consulta ao código Oracle e recupera todos os blocos de comentários das “procedures” e das “functions” dentro dos pacotes do projeto. Este conteúdo fica então disponível para a ferramenta. A partir daí é possível consultar o código recuperado, por representação de requisitos (p. ex.: representações de especificação de requisitos como os casos de uso e representações de regras de negócio), e também verificar se o padrão de instrumentação realizado manualmente no código está de acordo com o especificado. O resultado das duas consultas está representado como relatórios na Figura 20.

É possível concluir que a ferramenta automatiza as consultas ao código Oracle por representações de casos de uso, regras de negócio e contexto, o que torna a verificação da rastreabilidade mais confiável do que, por exemplo, uma matriz de rastreabilidade.

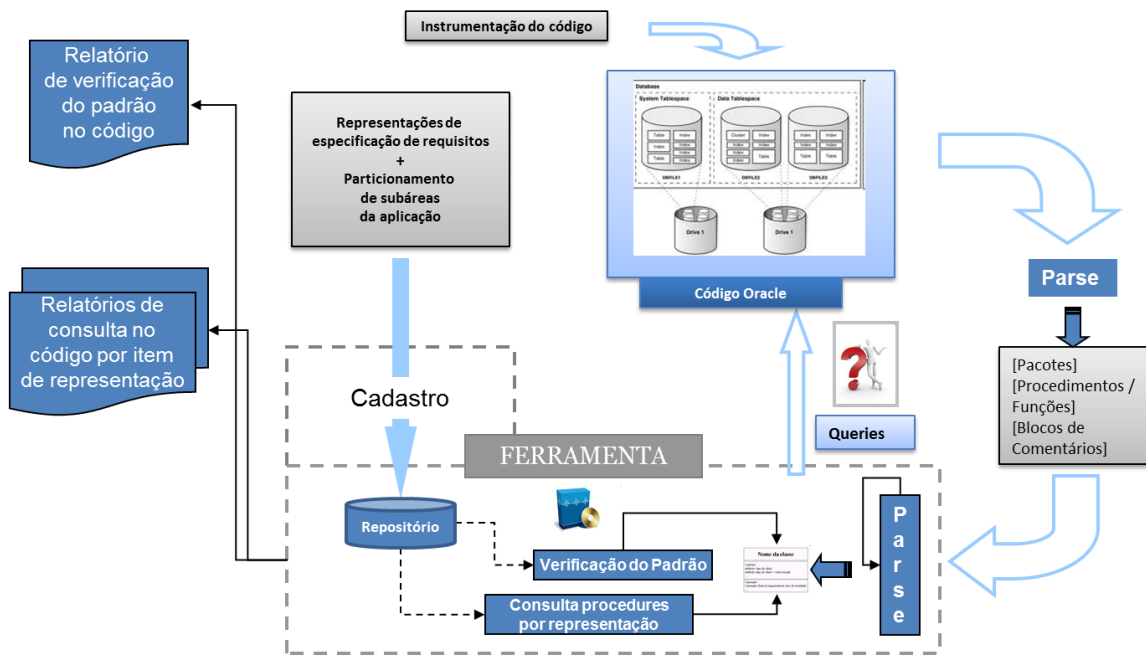


Figura 20 – Visão mais detalhada da ferramenta

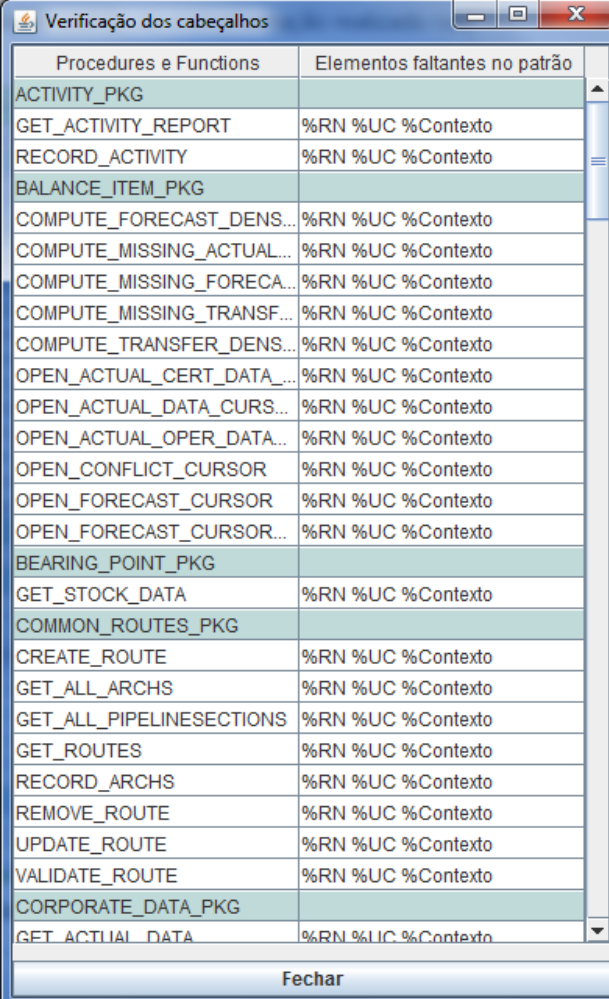
Quinta atividade: Testar.

Os testes da etapa de implementação consistem em definir alguns cenários de teste e verificar se a ferramenta apresenta um comportamento adequado como especificado.

A ferramenta deve possibilitar:

- 1- O cadastro de itens de representação de caso de uso;
- 2- Consulta ao código Oracle de acordo com uma representação de caso de uso.
- 3- A verificação da instrumentação feita no código Oracle para confirmar se a instrumentação foi realizada corretamente.

Os testes foram realizados no laboratório de engenharia de software (LES) do departamento de informática da PUC-RIO. Na Figura 21 abaixo, é apresentado o relatório de validação do padrão de instrumentação realizado nas “procedures” Oracle.



Procedures e Funcions	Elementos faltantes no padrão
ACTIVITY_PKG	
GET_ACTIVITY_REPORT	%RN %UC %Contexto
RECORD_ACTIVITY	%RN %UC %Contexto
BALANCE_ITEM_PKG	
COMPUTE_FORECAST_DENS...	%RN %UC %Contexto
COMPUTE_MISSING_ACTUAL...	%RN %UC %Contexto
COMPUTE_MISSING_FORECA...	%RN %UC %Contexto
COMPUTE_MISSING_TRANSF...	%RN %UC %Contexto
COMPUTE_TRANSFER_DENS...	%RN %UC %Contexto
OPEN_ACTUAL_CERT_DATA_...	%RN %UC %Contexto
OPEN_ACTUAL_DATA_CURS...	%RN %UC %Contexto
OPEN_ACTUAL_OPER_DATA...	%RN %UC %Contexto
OPEN_CONFLICT_CURSOR	%RN %UC %Contexto
OPEN_FORECAST_CURSOR	%RN %UC %Contexto
OPEN_FORECAST_CURSOR...	%RN %UC %Contexto
BEARING_POINT_PKG	
GET_STOCK_DATA	%RN %UC %Contexto
COMMON_ROUTES_PKG	
CREATE_ROUTE	%RN %UC %Contexto
GET_ALL_ARCHS	%RN %UC %Contexto
GET_ALL_PIPELINESECTIONS	%RN %UC %Contexto
GET_ROUTES	%RN %UC %Contexto
RECORD_ARCHS	%RN %UC %Contexto
REMOVE_ROUTE	%RN %UC %Contexto
UPDATE_ROUTE	%RN %UC %Contexto
VALIDATE_ROUTE	%RN %UC %Contexto
CORPORATE_DATA_PKG	
GET_ACTUAL_DATA	%RN %UC %Contexto

Fechar

Figura 21 – Relatório de validação da instrumentação

Na Figura 22 é apresentada uma consulta realizada pela aplicação, onde é verificado quais procedures e functions estão associadas ao caso de uso UC01.

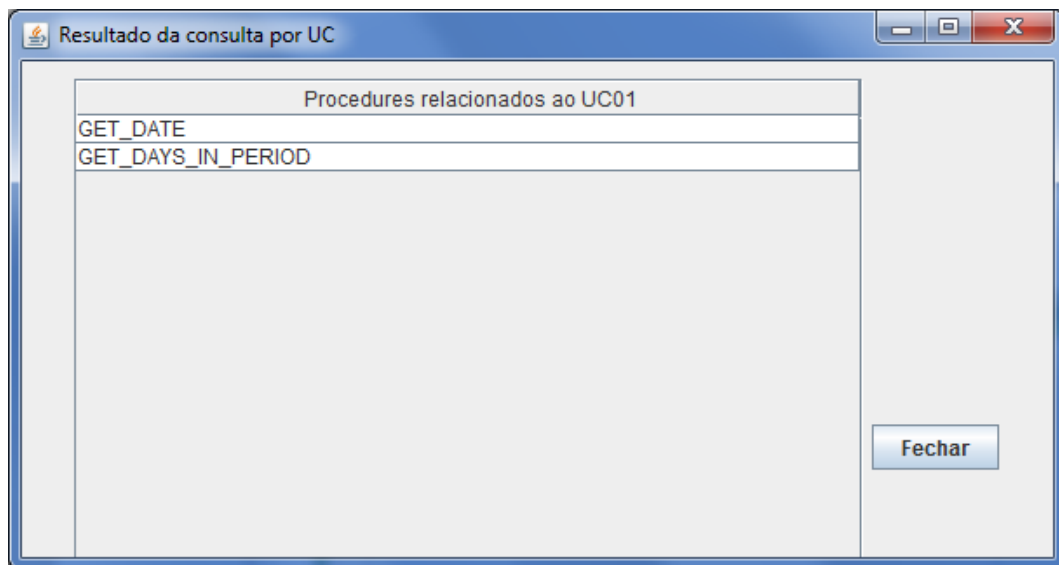


Figura 22 – Consulta às procedures Oracle por Caso de Uso

4.5. Instanciação: Validar Estratégia

As seguintes atividades compõem a etapa Validar Estratégia:

- 1- Validar arquitetura;
- 2- Avaliar usos em relação ao processo de software;
- 3- Avaliar arquitetura em relação aos usos.

4.5.1. Primeira atividade: Validar arquitetura.

Esta atividade compreende três tarefas:

- ✓ Identificar artefatos ambíguos;
- ✓ Identificar rastros voláteis;
- ✓ Identificar elos de rastreabilidade duplicados.

Para completar esta atividade, é necessário o suporte do documento de arquitetura elaborado na segunda etapa do processo. Desta forma é possível verificar se existe algum dos acima citados.

Em relação ao projeto utilizado nesta instanciação do processo, nenhum item listado acima foi identificado.

4.5.2. Segunda atividade: Avaliar usos em relação ao processo de software.

Esta atividade compreende três tarefas:

- ✓ Identificar os usos de rastreabilidade ausentes;
- ✓ Verificar se todos os atores que necessitam de rastreabilidade estão contemplados;
- ✓ Identificar usos de rastreabilidade supérfluos.

Para executar todas as tarefas desta atividade, é necessário realizar uma comparação entre o modelo de uso construído na etapa de projeto com as metas dos atores do processo de software. As metas dos atores do processo de software devem ser coletadas assim como as suas tarefas.

Em relação ao projeto utilizado nesta instanciação do processo, nenhuma pendência foi identificada nesta atividade.

4.5.3. Terceira atividade: Avaliar arquitetura em relação aos usos.

Esta atividade compreende duas subatividades:

- ✓ Identificar elos de rastreabilidade ausentes;
- ✓ Identificar elos de rastreabilidade supérfluos.

Para executar todas as tarefas desta atividade, é necessário comparar o modelo de uso com a arquitetura de rastreabilidade, ambos elaborados na segunda etapa do processo.

Em relação à instanciação do processo realizada neste trabalho, foi possível verificar que os elos de rastreabilidade atendem as necessidades do projeto.

Ao final desta atividade é possível identificar se foi construída uma rastreabilidade utilizável e se as estratégias de rastreabilidade aplicadas suportam todas as necessidades de rastreabilidade específicas do projeto.

4.6.

Instanciação: Utilizar

Nesta etapa os atores de uso da estratégia de rastreabilidade utilizam os rastros construídos nas etapas anteriores. Nesta etapa, a percepção de novas necessidades pode ser descrita e encaminhada aos atores responsáveis pela manutenção da estratégia.

4.7.

Uso do ferramental para construção dos requisitos

4.7.1. Desenho da ferramenta

Para a construção da aplicação descrita neste trabalho, foi realizada inicialmente na concepção do projeto, uma busca de artigos sobre rastreabilidade em ambientes PL/SQL, acesso ao código de pacotes e procedures Oracle. Um estudo preliminar foi realizado sobre o ambiente de desenvolvimento de procedures PL/SQL e as relações entre os elementos do SGBD Oracle, além de discussões com programadores PL/SQL do referido sistema.

O projeto foi iniciado com o levantamento dos requisitos básicos para o desenvolvimento de uma ferramenta que apoiasse a construção da rastreabilidade, principalmente entre casos de uso e procedures Oracle. A seguir foram feitas algumas pesquisas sobre como realizar consultas às procedures Oracle e também sobre formas de se obter a rastreabilidade de requisitos e procedures Oracle. Para isso, foram utilizados manuais da Oracle impresso e documentação Oracle disponível na web. Foi elaborado um escopo inicial da ferramenta com base no tempo disponível para o projeto.

Com base no escopo inicial de requisitos, foi iniciado o processo de modelagem das classes que suportariam a aplicação e elaboração dos padrões de projeto. A seguir foi elaborado um padrão de instrumentação das procedures Oracle. O termo “cabeçalho” empregado neste na descrição da ferramenta se refere ao bloco de comentário que é incluído antes do código das procedures Oracle.

Inicialmente, foi elaborado um padrão de instrumentação mais abrangente, de forma a contemplar várias informações sobre a implementação da procedure e sua manutenção ao longo do ciclo de vida do sistema. Este padrão apresentou uma série de problemas no tratamento do espaço de memória consumida na

busca por expressões regulares (regex). Posteriormente, foi elaborado um padrão de instrumentação adaptando o padrão de comentário para o javadoc da ferramenta do RSA (Rational Software Architect) da IBM Rational, que foi utilizado como modelo. Desta forma, foi incluído dentro dos cabeçalhos, um padrão de instrumentação mais restrito e que suportasse a busca pelos requisitos.

O termo “requisito” no contexto da ferramenta refere-se inicialmente a uma representação ou a um conjunto de representações da especificação dos requisitos do software de uso das procedures Oracle. Por exemplo, o código “UC01” é uma representação do documento de caso de uso UC01 utilizado na especificação do software.

O conteúdo principal do padrão foi elaborado para suportar, a rastreabilidade entre as procedures Oracle e o conjunto de representação. Esse conjunto de representação foi utilizado de forma mais abrangente, para caracterizar além dos documentos de especificação de casos de uso e itens do documento de regras de negócio, também os itens de contexto do sistema, recuperados do diagrama conceitual do software.

A seguir é apresentado um exemplo do padrão de instrumentação elaborado para as procedures Oracle:

```
1 * <!-- begin-Trace -->
2   %UC: UC01
3   %RN: RN53
4   %Domain: OS Entrega para Terceiros
5 * <!-- end-Trace -->
```

Figura 23 – Padrão de instrumentação

É possível verificar que nas linhas 1 e 5 na Figura 20, o início e o fim do texto padrão de instrumentação inserido nos cabeçalhos. Nas linhas 2, 3 e 4, encontra-se a parte principal do texto, que no exemplo acima, foi utilizada para realizar a rastreabilidade entre uma procedure Oracle e (1) a representação de especificação UC01(Caso de Uso 01), (2) a regra de negócio RN53 e (3) o item de contexto do sistema “Operação de Suprimento Entrega para Terceiros” (referente à funcionalidade do sistema “OS Entrega para Terceiros”).

A aplicação consegue detectar uma especificação de requisito - sendo que, no contexto da instanciação descrita neste capítulo, considera-se uma

representação de uma especificação de um caso de uso - presente no padrão de instrumentação se o mesmo foi cadastrado no cadastro de itens de representações de requisitos. Possíveis falhas de cadastro de requisitos ou no preenchimento do padrão de instrumentação nas procedures Oracle, não são detectadas pela aplicação.

4.7.2. Arquitetura

A arquitetura do sistema foi idealizada em quatro pacotes de acordo com a Figura 19. O pacote **GUI** representa as classes de interface, o pacote **Connection** representa as classes responsáveis pela conexão ao banco de dados Oracle, o **Fachadas** representa as classes facates e o **Domain** as classes de domínio.

Na Figura 21 é apresentada a arquitetura desenvolvida para essa aplicação.

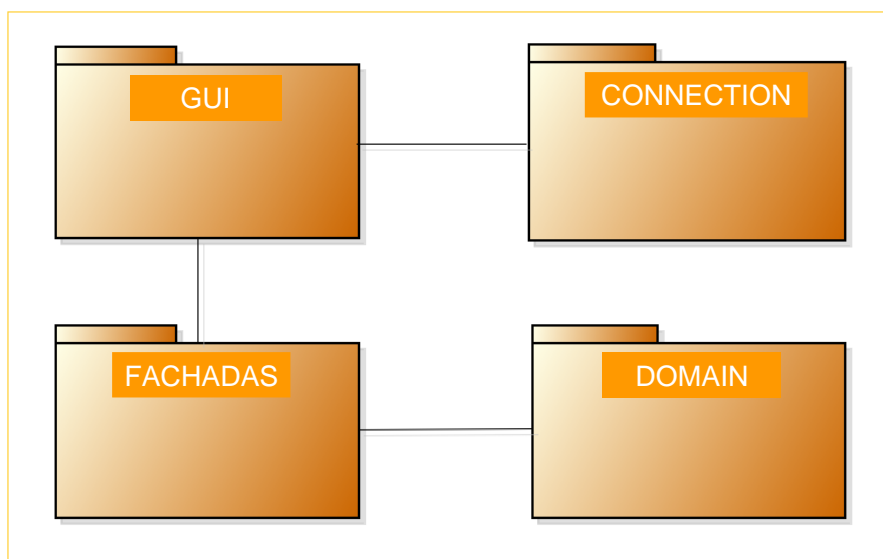


Figura 24 – Arquitetura da aplicação

No Apêndice I e no Apêndice II, são apresentados respectivamente os algoritmos mais relevantes e os diagramas de classes da ferramenta.

No capítulo seguinte são apresentadas as conclusões deste trabalho.