

UNIVERSIDADE FEDERAL DE SÃO CARLOS

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**ESPECIFICAÇÃO DE REQUISITOS NO DOMÍNIO DE
SISTEMAS DE INFORMAÇÃO COM O USO DE
PADRÕES**

LEONARDO VIEIRA BARCELOS

ORIENTADORA: PROF. DRA. ROSÂNGELA A. D. PENTEADO

São Carlos - SP
Janeiro/2016

UNIVERSIDADE FEDERAL DE SÃO CARLOS

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**ESPECIFICAÇÃO DE REQUISITOS NO DOMÍNIO DE
SISTEMAS DE INFORMAÇÃO COM O USO DE
PADRÕES**

LEONARDO VIEIRA BARCELOS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação, área de concentração: Engenharia de Software.

Orientadora: Dra. Rosângela A. D. Penteado

São Carlos - SP
Janeiro/2016

Ficha catalográfica elaborada pelo DePT da Biblioteca Comunitária UFSCar
Processamento Técnico
com os dados fornecidos pelo(a) autor(a)

B242e Barcelos, Leonardo Vieira
Especificação de requisitos no domínio de sistemas de informação com o uso de padrões / Leonardo Vieira Barcelos. -- São Carlos : UFSCar, 2016.
108 p.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2016.

1. Documento de requisitos. 2. Padrões de requisitos. 3. Reúso de requisitos. 4. Domínio de sistemas de informação. 5. Apoio computacional com base em padrões de requisitos. I. Título.



UNIVERSIDADE FEDERAL DE SÃO CARLOS
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Folha de Aprovação

Assinaturas dos membros da comissão examinadora que avaliou e aprovou a defesa de dissertação de mestrado do candidato Leonardo Vieira Barcelos, realizada em 03/03/2016.

Profa. Dra. Rosângela Aparecida Delosso Penteado
(UFSCar)

Profa. Dra. Rosane Teresinha Vaccare Braga
(ICMC/USP)

Prof. Dr. Paulo Afonso Parreira Júnior
(UFLA)

Certifico que a sessão de defesa foi realizada com a participação à distância do membro Prof. Dr. Paulo Afonso Parreira Júnior e, depois das arguições e deliberações realizadas, o participante à distância está de acordo com o conteúdo do parecer da comissão examinadora redigido no relatório de defesa do aluno Leonardo Vieira Barcelos.

Profa. Dra. Rosângela Aparecida Delosso Penteado
Presidente da Comissão Examinadora
(UFSCar)

AGRADECIMENTO

Primeiramente, agradeço a Deus pela oportunidade de fazer um curso de mestrado.

À professora Dra. Rosângela por ter me escolhido como seu orientando. Obrigado pela amizade, pela confiança e por me ajudar a evoluir como pesquisador.

Aos professores Dra. Rosana Teresinha Vaccare Braga e Dr. Valter Vieira de Camargo pelas sugestões para a melhoria deste trabalho.

A minha querida esposa Francielle, pelas palavras e gestos de compreensão e apoio que se fizeram presentes em todos os momentos deste trabalho.

Aos meus pais e familiares por toda demonstração de amor, incentivo e confiança.

Ao Ivan Reis, meu amigo, companheiro de mestrado e viagem para São Carlos, meus sinceros agradecimentos.

Aos amigos Gilberto Trivelato e Alessandro Santana, meus agradecimentos pelo incentivo e motivação durante a realização deste trabalho.

A UFSCar e aos professores, pelos conhecimentos adquiridos.

RESUMO

Estudos apontam que as principais causas de falhas em projetos de software estão relacionadas ao documento de especificação de requisitos que contém requisitos incompletos, inconsistentes e ambíguos. O desenvolvimento de software com qualidade requer uma especificação de requisitos completa e consistente. Um dos problemas enfrentados pelos engenheiros de software é que, algumas vezes, eles não conhecem completamente o domínio da aplicação e têm dificuldade de comunicação com os *stakeholders* para a especificação dos requisitos. O reúso de conhecimentos obtidos de projetos concluídos pode facilitar a identificação e a escrita de requisitos na elaboração de um documento de requisitos completo e consistente. No domínio de sistemas de informação é comum encontrar um conjunto de requisitos que possuem características similares e que está presente em diferentes sistemas, o que indica um possível padrão de requisito. A vantagem no uso de padrões de requisitos é propiciar ao engenheiro de software um ponto de partida para a elaboração do documento de requisitos, em vez de começar do zero para resolver problemas conhecidos. Os padrões de requisitos podem contribuir para a melhoria da qualidade do software, minimizar o tempo e o custo do desenvolvimento, uma vez que são reusadas soluções previamente definidas e testadas. Nesta dissertação, foi desenvolvido um conjunto de padrões de requisitos para auxiliar o engenheiro de software na elaboração do documento de requisitos de sistemas de informação. A fim de facilitar e padronizar o documento de requisitos elaborado, foi também desenvolvido um apoio computacional com base nesse conjunto de padrões. Estudos de casos com estudantes de graduação da UFSCar e UEMG foram utilizados para a avaliação dos padrões e do apoio computacional, objetivando comprovar a eficácia desta proposta, bem como identificar possíveis melhorias. Ao analisar os estudos de caso realizados, pôde-se inferir que há indícios que os padrões elaborados auxiliaram na escrita de um documento de requisitos mais completo, que o apoio computacional é de fácil utilização e que houve aumento de produtividade por parte, principalmente, dos desenvolvedores inexperientes.

Palavras-chave: documento de requisitos, padrões de requisitos, reúso de requisitos, domínio de sistemas de informação, apoio computacional com base em padrões de requisitos.

ABSTRACT

Studies show that the main causes of failures in software projects are related to the requirements specification document that contains incomplete, inconsistent and ambiguous requirements. The software development with quality requires a complete and consistent requirements specification. One of the problems faced by software engineers is that, sometimes, they do not know completely the application domain and have difficulty in communication with stakeholders to obtain the requirements specification. The knowledge reuse obtained from completed projects can facilitate the identification and writing of requirements in the elaboration of a complete and consistent requirements document. In the information systems domain it is common to find a set of requirements that has similar characteristics, which is present in different systems, indicating a possible requirement pattern. The advantage of using requirements patterns is provide to the software engineer a starting point for the elaboration of the requirements document, instead of starting from scratch to solve known problems. The requirements patterns can contribute to the improvement of software quality, minimizing the development time and cost, because previously defined and tested solutions are reused. In this dissertation it was developed a set of requirements patterns to assist the software engineer in the elaboration of requirements document of information systems. In order to facilitate and standardize the requirements document elaborated it was also developed a computational support based in that set of patterns. Case studies were used to evaluate the patterns and the computational support, with graduate students of UFSCar and UEMG, aiming prove the effectiveness of this proposal, as wellas to identify possible improvements. Analyzing the case studies conducted, it can be inferred that there is evidence that the elaborated patterns assist in the elaboration of a more complete requirements document, that the computational support is easy to use, and that there was an increased productivity, mainly of inexperienced developers.

Keywords: requirements document, requirements patterns, requirements reuse, system information domain, computational support based on requirements patterns.

LISTA DE FIGURAS

Figura 2.1 – Processo de Engenharia de Requisitos (SOMMERVILLE, 2007)	20
Figura 2.2 – Catálogo de Padrões de Requisitos de Software (WITHALL, 2007)	22
Figura 2.3 – Linguagem de Padrões para Gestão de Recursos de Negócios (BRAGA; GERMANO; MASIERO, 1999)	28
Figura 2.4 – Padrão Identificar o Recurso (BRAGA; GERMANO; MASIERO, 1999)	29
Figura 2.5 – Instanciação do Padrão Identificar o Recurso (BRAGA; GERMANO; MASIERO, 1999).....	29
Figura 3.1 – Metodologia para Elaboração dos Padrões	36
Figura 3.2 – Padrões de Requisitos para o Domínio de Sistemas de Informação	47
Figura 4.1 – Modelo Conceitual do Apoio Computacional.....	50
Figura 4.2 – Módulo de Manutenção de Usuário.....	53
Figura 4.3 – Módulo de Manutenção de Cliente.....	53
Figura 4.4 – Módulo de Manutenção de Projeto.....	54
Figura 4.5 – Módulo de Manutenção de Sugestões de Parâmetros.....	55
Figura 4.6 – Módulo de Especificação e Gestão de Padrões.....	56
Figura 4.7 – Módulo de Instanciação de Padrões – Aba Informações Básicas.....	57
Figura 4.8 – Módulo de Instanciação de Padrões – Aba Elaboração.....	58
Figura 4.9 – Módulo de Instanciação de Padrões – Aba Requisitos	59
Figura 4.10 – Módulo de Instanciação de Padrões – Aba Reúso.....	60
Figura 4.11 – Exemplo de Requisitos Especificados pelo Apoio Computacional	60
Figura 4.12 – Cadastro do Projeto de Videolocadora.....	62
Figura 4.13 – Seleção do Projeto de VideoLocadora	62
Figura 4.14 – Instanciação do Padrão Incluir Informação	63
Figura 4.15 – Escrita do Requisito para Incluir Informação de Filme: Preenchimento do Parâmetro <i>entidade</i>	65
Figura 4.16 – Escrita do Requisito para Incluir Informação de Filme: Preenchimento do Parâmetro <i>atributos</i>	65
Figura 4.17 – O Apoio Computacional Sugere o Uso de Padrões Relacionados.....	66
Figura 4.18 – Caixa de Diálogo com os Padrões Relacionados ao Padrão Incluir Informação	67
Figura 4.19 – Escrita do Requisito para Alterar Informação de Filme: Preenchimento do Parâmetro <i>entidade</i>	67

Figura 4.20 – Escrita do Requisito para Alterar Informação de Filme: Preenchimento do Parâmetro <i>atributos</i>	68
Figura 4.21 – Visualização de Requisitos de Projetos Concluídos.....	69
Figura 4.22 – Editando Requisito Reusado de Projetos Concluídos.....	69
Figura 4.23 – Alteração do Requisito Reusado de Projetos Concluídos.....	70
Figura 4.24 – Requisito Reusado para a Inclusão de Cliente.....	71
Figura 4.25 – Requisito para Alterar Informação de Cliente.....	71
Figura 4.26 – Requisito para a Consulta da Situação do Filme.....	72
Figura 4.27 – Escrita do Requisito Processar Transação de Empréstimo de Filme: Preenchimento do Parâmetro <i>transação</i>	73
Figura 4.28 – Escrita do Requisito Processar Transação de Empréstimo de Filme: Preenchimento do Parâmetro <i>entidade</i>	73
Figura 4.29 – Escrita do Requisito Processar Transação de Empréstimo de Filme: Preenchimento do Parâmetro <i>atributos</i>	74
Figura 4.30 – Requisito para o Processamento de Empréstimo de Filme.....	75
Figura 4.31 – Caixa de Diálogo com os Padrões Relacionados ao Padrão Processar Transação.....	75
Figura 4.32 – Escrita do Requisito Condição de Empréstimo: Preenchimento do Parâmetro <i>operação</i>	76
Figura 4.33 – Escrita do Requisito Condição de Empréstimo: Preenchimento do Parâmetro <i>condição</i>	76
Figura 4.34 – Escrita do Requisito Recuperar Informação de Reserva: Preenchimento do Parâmetro <i>entidade</i>	77
Figura 4.35 – Escrita do Requisito Recuperar Informação de Reserva: Preenchimento do Parâmetro <i>atributos</i>	77
Figura 4.36 – Escrita do Requisito Recuperar Informação de Reserva: Preenchimento do Parâmetro <i>condição</i>	78
Figura 4.37 – Requisitos do Projeto de Videolocadora.....	78
Figura 4.38 – Escrita do Requisito Processar Transação de Reserva: Preenchimento do Parâmetro <i>transação</i>	79
Figura 4.39 – Escrita do Requisito Processar Transação de Reserva: Preenchimento do Parâmetro <i>entidade</i>	80
Figura 4.40 – Escrita do Requisito Processar Transação de Reserva: Preenchimento do Parâmetro <i>atributos</i>	80
Figura 4.41 – Requisito para a Impressão de Relatório de Empréstimos de Filme da Data Atual.....	81
Figura 4.42 – Documento de Requisitos Elaborado com o Apoio Computacional.....	82

LISTA DE QUADROS

Quadro 2.1 – Padrão: Communicating Resource Consumption (ROHER; RICHARDSON, 2013)	25
Quadro 2.2 – Conteúdo do Documento de Especificação de Requisitos (IEEE, 1998)	31
Quadro 3.1 – Estrutura Adotada para Apresentação dos Padrões	38
Quadro 3.2 – Catálogo de Padrões de Requisitos para o Domínio de Sistemas de Informação	39
Quadro 3.3 – Padrão Incluir Informação	40
Quadro 3.4 – Padrão Alterar Informação	41
Quadro 3.5 – Padrão Excluir Informação	41
Quadro 3.6 – Padrão Recuperar Informação	42
Quadro 3.7 – Padrão Manter Informação	42
Quadro 3.8 – Padrão Processar Transação	43
Quadro 3.9 – Padrão Especificar Consulta Gerencial	44
Quadro 3.10 – Padrão Condição de Execução	45
Quadro 3.11 – Padrão Cálculo de Valor	46
Quadro 3.12 – Padrão Informação Requerida	46
Quadro 3.13 – Padrão Limite de Execução	47
Quadro 4.1 – Visão Geral do Sistema de Videolocadora	61

LISTA DE ABREVIATURAS E SIGLAS

DR – Documento de Requisitos.

ER – Engenharia de Requisitos.

ERS – Especificação de Requisitos de Software.

PDF – *Portable Document Format* – Formato de documento portátil.

PRS – Padrão de Requisito de Software.

SAD – Sistemas de Apoio a Decisões.

SI – Sistemas de Informação.

SIG – Sistemas de Informações Gerenciais.

SPT – Sistemas de Processamento de Transação.

UML – *Unified Modeling Language* – Linguagem de modelagem unificada.

SUMÁRIO

CAPÍTULO 1 - INTRODUÇÃO.....	12
1.1 Contexto.....	12
1.2 Motivação.....	14
1.3 Objetivo.....	16
1.4 Organização da Dissertação.....	17
CAPÍTULO 2 - FUNDAMENTAÇÃO TEÓRICA.....	18
2.1 Considerações Iniciais.....	18
2.2 Requisitos de Software.....	19
2.3 Padrões de Requisitos.....	21
2.4 Trabalhos Relacionados.....	23
2.5 Conceitos em Sistemas de Informação.....	26
2.5.1 Reúso no Desenvolvimento de Sistemas de Informação.....	28
2.6 A norma IEEE Std 830.....	30
2.7 Considerações Finais.....	32
CAPÍTULO 3 - PADRÕES DE REQUISITOS FUNCIONAIS PARA SISTEMAS DE INFORMAÇÃO.....	34
3.1 Considerações Iniciais.....	34
3.2 Metodologia de Elaboração.....	35
3.3 Padrões de Requisitos Elaborados.....	39
3.3.1 Padrões de Requisitos: Operações Básicas.....	40
3.3.2 Padrão de Requisito: Processamento de Transação para SI.....	43
3.3.3 Padrão de Requisito: Informações Gerenciais para SI.....	44
3.3.4 Padrões de Requisito: Regras de Negócio para SI.....	45
3.4 Considerações Finais.....	48
CAPÍTULO 4 - APOIO COMPUTACIONAL PARA A ELABORAÇÃO DE DOCUMENTO DE REQUISITOS COM PADRÕES.....	49
4.1 Considerações Iniciais.....	49
4.2 Construção do Apoio Computacional.....	50
4.3 Processo de Uso do Apoio Computacional.....	51

4.4 Funções Implementadas no Apoio Computacional	52
4.5 Exemplo de Aplicação do Apoio Computacional	61
4.5.1 Elaboração do Documento de Requisitos	61
4.6 Considerações Finais	83
CAPÍTULO 5 - ESTUDOS DE CASO	84
5.1 Considerações Iniciais.....	84
5.2 Estudo de Caso 1	85
5.3 Estudo de Caso 2.....	88
5.4 Estudo de Caso 3	90
5.5 Estudo de Caso 4.....	92
5.6 Considerações Finais	94
CAPÍTULO 6 - CONSIDERAÇÕES FINAIS.....	96
6.1 Resultados Obtidos	96
6.2 Limitações da Proposta.....	99
6.3 Trabalhos Futuros	99
REFERÊNCIAS.....	101
APÊNDICE A.....	103
APÊNDICE B	106

Capítulo 1

INTRODUÇÃO

1.1 Contexto

Problemas relacionados com a Engenharia de Requisitos (ER) são amplamente reconhecidos por afetar a qualidade do software e impactar a eficácia do seu processo de desenvolvimento (NIAZI; SHASTRY, 2003). Estima-se que descobrir e corrigir um problema após a entrega do software, pode ser 100 (cem) vezes mais caro do que se essa correção ocorrer durante as fases iniciais do desenvolvimento (BOEHM; BASILI, 2001). Nesse sentido, há relatos de que a completa compreensão e especificação de requisitos está entre as tarefas mais difíceis enfrentadas por um engenheiro de software (PRESSMAN, 2011).

Chernak (2012) relata que na indústria de TI existem três formas principais de documentação de requisitos: a ER tradicional (48%), a dirigida por casos de uso (28%), a de métodos ágeis (11%) e as demais abordagens (13%). Observa-se que a ER tradicional corresponde à principal forma para a documentação de requisitos. Contudo, normalmente os requisitos são especificados usando linguagem natural, o que pode resultar em problemas como ambiguidade e incompletude (FATWANTO, 2013). Assim constata-se a necessidade de pesquisas que abordem a especificação de requisitos no contexto da ER tradicional a fim de melhorar a escrita do documento de requisitos (DR).

Dentre os problemas comuns na elicitação de requisitos pode-se citar a dificuldade dos *stakeholders* em expressar suas necessidades e a dos engenheiros de software, sem experiência ou amplo conhecimento do domínio do cliente em entendê-las completamente (SOMMERVILLE, 2007). Geralmente, o produto de

software desenvolvido não atende às reais necessidades dos *stakeholders*, gerando retrabalho, aumento do custo, a insatisfação ou até o cancelamento do projeto.

O reúso de software é uma solução reconhecida como um meio de melhorar a qualidade do software e de minimizar o tempo e custo no desenvolvimento de software (KETABCHI; SANI; LIU, 2011). Além disso, o reúso propicia o aproveitamento de conhecimentos adquiridos em projetos anteriores, o que aumenta as chances de sucesso de novos projetos (PALOMARES et al., 2013).

Padrões de software têm sido usados nas mais variadas áreas, para captar e reutilizar conhecimento de diferentes contextos no desenvolvimento de software, como na arquitetura, no projeto de componentes, na implementação e também na engenharia de requisitos, que é o enfoque deste trabalho. Um dos primeiros trabalhos sobre padrões de requisitos de software é o de Withall (2007), que apresenta um catálogo com trinta e sete Padrões de Requisitos de Software (PRS) em oito domínios, que favorecem a escrita de requisitos.

Um PRS corresponde a um artefato que fornece orientações sobre como especificar requisitos funcionais, requisitos não funcionais e regras de negócio, de forma que possam ser reusados em contextos e problemas bem definidos.

Um padrão possui uma estrutura de apresentação pela qual é possível captar, documentar as informações e reutilizar o conhecimento expresso nesse padrão. Há várias formas de se apresentar um padrão. Em geral, quatro elementos são essenciais: nome do padrão (descrição geral da aplicabilidade), problema (o que pretende resolver), solução (descrição de como obter o resultado desejado) e consequências (implicações com o uso do padrão) (GAMMA et al., 1995).

Os padrões de requisitos também orientam o engenheiro de software na identificação e escrita de requisitos, fornecendo sugestões de informações que devem ser consideradas para a elaboração completa e consistente de um DR.

A elaboração de padrões pode ser realizada por meio de conhecimentos adquiridos a partir de pesquisas de campo, do conhecimento individual de engenheiros de software, de normas existentes e/ou de melhores práticas ou quaisquer outros artefatos de software (ROHER; RICHARDSON, 2013).

1.2 Motivação

A reutilização de software pode contribuir para o aumento da produtividade, uma vez que conduz: a) à melhoria do tempo de colocação do produto no mercado; b) à melhoria da qualidade do produto; c) a não iniciação de um desenvolvimento do zero; e d) à redução do custo de desenvolvimento (CHERNAK, 2012). A aplicação de abordagens de reuso desde a fase inicial do projeto pode ajudar os engenheiros de software na elicitação, validação e documentação dos requisitos do software, podendo ter como consequência uma especificação mais completa, consistente e sem ambiguidade (KETABCHI; SANI; LIU, 2011) (PALOMARES; QUER; FRANCH, 2011).

O uso de padrões de software auxilia engenheiros de software na reutilização de soluções de sucesso para desenvolver novos projetos (GAMMA et al., 1995). Na ER, padrões de requisitos podem ser usados tanto na identificação como também na escrita dos requisitos de um DR, eliminando a redundância de definição de um conjunto de requisitos comum para projetos de software. Dessa forma, pode haver redução da carga de trabalho, pois existe um roteiro a ser seguido, o que facilita a comunicação entre os *stakeholders*.

No desenvolvimento de software existem requisitos que são de natureza similar ou que aparecem com frequência na maioria dos softwares, o que indica um possível padrão (WITHALL, 2007). Essa similaridade pode ser encontrada em requisitos de sistemas de diversos domínios quanto aos requisitos funcionais que descrevem a entrada e armazenamento de dados. No domínio de Sistemas de Informação (SI), além desses, é comum encontrar requisitos funcionais comuns para o processamento de transações, relatórios ou consultas de informações gerenciais e regras de negócio. Assim, a elaboração de padrões de requisitos no domínio de SI pode proporcionar ao engenheiro de software a reutilização de experiências bem-sucedidas que auxiliem na identificação e escrita de requisitos para a elaboração de um DR completo e consistente, resultando no aumento da qualidade do sistema em desenvolvimento.

Com um conjunto de padrões de requisitos elaborados com esse objetivo, neste projeto chamado de catálogo de padrões, o engenheiro de software passa a ter um ponto de partida para a especificação dos requisitos e a comunicação com os *stakeholders*, em vez de começar do zero para resolver problemas conhecidos (PALOMARES et al., 2013).

Em função da evidente importância e os problemas enfrentados no processo de ER, observa-se a existência de algumas lacunas de pesquisa em relação à elaboração do DR, para que o mesmo seja completo e consistente. A reutilização de requisitos com o uso de padrões pode ser uma alternativa viável para aumentar a qualidade da especificação dos requisitos e a produtividade no desenvolvimento de software.

Um catálogo de padrões de requisitos no domínio de SI também contribui para a identificação dos requisitos. Durante a elicitação o engenheiro de software, de acordo com a necessidade declarada pelos *stakeholders*, seleciona no catálogo o padrão que melhor atende ao problema citado e pode também solicitar mais detalhes específicos da solução. Além disso, para complementar a solução do problema apresentada para cada padrão, há sugestões de informações que são fornecidas por meio de padrões relacionados. Um importante elemento que deve fazer parte da estrutura de apresentação de um padrão de requisito são os padrões relacionados, objetivando a completude da especificação. O engenheiro de software também pode sugerir o uso de padrões que contemplem as funcionalidades previamente declaradas pelos *stakeholders*.

Cada padrão de requisito fornece um *template* como solução para escrita de requisitos que atende a um problema específico. Nesse *template*, geralmente o texto possui uma parte fixa e outra variável. A parte fixa representa a operação a ser realizada e a parte variável os detalhes específicos dessa operação, como: a entidade de dados atendida, os atributos necessários, restrições específicas do negócio, entre outros.

Mesmo com o catálogo de padrões pode ocorrer dificuldade para o desenvolvedor de software utilizar ou elencar todos os padrões que necessita. Assim, é interessante ter um apoio computacional com base nesses padrões, de modo que facilite a busca e seleção de padrões, forneça sugestões para os detalhes específicos da solução do padrão e também proporcione a reutilização de requisitos que foram escritos com padrões em projetos concluídos, aumentando as possibilidades de reúso. Esse apoio computacional proporciona ao engenheiro de software todo o subsídio necessário para a elaboração de um DR, além de uniformizar as informações uma vez que há um guia a ser seguido.

Uma revisão sistemática da literatura (RSL) foi conduzida para identificar os padrões de requisitos existentes e que podem contribuir para a melhoria da qualidade

de DR. A maioria dos trabalhos encontrados aborda padrões para requisitos não funcionais, que podem ser reutilizados em diferentes sistemas por não tratarem de particulares específicas do sistema em desenvolvimento. Também foram encontrados alguns padrões para requisitos funcionais para sistemas de gestão de documentos. Entretanto, nem sempre esses padrões estavam disponíveis.

Para o desenvolvimento deste trabalho restringiu-se à elaboração de padrões de requisitos para atender ao domínio de SI, por corresponder a várias aplicações de software para a operacionalização e gestão de empresas.

1.3 Objetivo

Este trabalho tem por objetivo elaborar um conjunto de padrões de requisitos funcionais e de regras de negócio para o domínio de SI, bem como um apoio computacional desenvolvido com base nesses padrões para a escrita de um DR.

O apoio computacional elaborado deve amparar o engenheiro de software durante a elaboração do DR com o uso dos padrões de modo a:

- Permitir a especificação e gestão dos padrões de requisitos;
- Permitir o estabelecimento de relacionamento entre os padrões;
- Permitir a instanciação de padrões de requisitos, fornecendo sugestões dos detalhes específicos da solução, como por exemplo os atributos necessários em um cadastro de cliente;
- Permitir que os requisitos especificados em projetos concluídos com a utilização dos padrões, sejam disponibilizados em projetos em desenvolvimento com a possibilidade de reutilização e também de serem alterados;
- Permitir que os padrões relacionados sejam sugeridos durante a instanciação de padrões;
- Permitir a definição de relacionamento de dependência entre os requisitos instanciados.

Para avaliar os padrões e o apoio computacional, estudos de casos foram conduzidos para verificar a eficácia da proposta, bem como a usabilidade do apoio computacional.

As principais contribuições deste trabalho são:

- a) Melhorar a especificação de DRs no domínio de SI por meio de padrões de requisitos funcionais e de regras de negócio;
- b) Um catálogo de padrões para a identificação e para a escrita de requisitos no domínio de SI;
- c) Um apoio computacional com base no conjunto de padrões que favorece a escrita completa e consistente dos requisitos obtidos dos *stakeholders*, gerando um DR padronizado e com instâncias específicas do sistema em elaboração;
- d) Um repositório de requisitos que foi elaborado a partir dos padrões descritos em a);
- e) A possibilidade de escrita de novos padrões de requisitos a partir de uma estrutura elaborada;
- f) A disponibilização dos padrões de requisitos e do apoio computacional para a comunidade;
- g) A apresentação dos resultados dos estudos de casos que abordaram a aplicação dos padrões e do apoio computacional.

1.4 Organização da Dissertação

Este trabalho está organizado em 6 capítulos. Neste primeiro foi feita a contextualização do problema, a apresentação da motivação e dos objetivos para o presente trabalho.

No Capítulo 2 é citada a fundamentação teórica dos assuntos relacionados com o presente trabalho. No Capítulo 3 são apresentados os padrões de requisitos elaborados para o domínio de sistemas de informação.

No Capítulo 4 é apresentado o apoio computacional desenvolvido para a elaboração de documentos de requisitos com base nos padrões exibidos no Capítulo 3, bem como exemplos de utilização.

No Capítulo 5 são apresentados os resultados da avaliação do uso do apoio computacional e dos padrões elaborados.

No Capítulo 6 são comentadas as considerações finais e sugestões de trabalhos futuros.

Capítulo 2

FUNDAMENTAÇÃO TEÓRICA

2.1 Considerações Iniciais

A obtenção dos requisitos é o ponto de partida para o desenvolvimento de um sistema de software. A solução de inconsistências e erros de requisitos nas fases iniciais de desenvolvimento possibilita produzir um software de qualidade, com custo reduzido e entrega no prazo programado (FERREIRA; SILVA, 2009).

O uso de padrões de requisitos para a elaboração de um DR é uma alternativa viável para o reúso de conhecimento e de experiências de engenheiros de software. Esses padrões contribuem para que o DR seja completo e consistente, aumentando a qualidade do produto de software desenvolvido. Além disso, com a utilização de um apoio computacional com base nesses padrões, há um aumento da produtividade porque há um guia a ser seguido para a identificação e escrita dos requisitos, evitando a escrita a partir do zero, uma vez que são reusadas soluções comprovadas e testadas.

Os requisitos funcionais tendem a ser mais específicos e geralmente a sua reutilização fica restrita a um único domínio. Neste trabalho, o domínio escolhido para a elaboração de padrões de requisitos funcionais foi o de SI, que possuem funções relacionadas à operacionalização e gestão de negócios, como a manutenção de dados, o processamento de transações e a emissão de relatórios gerenciais. Além disso, esse domínio é bem conhecido por ser a maior área de aplicação de software. Ressalta-se que a entrada e manutenção de dados é comum na maioria dos sistemas, sendo que os padrões com essa finalidade podem ser usados para qualquer domínio de software.

Para a elaboração dos padrões de requisitos e o desenvolvimento do apoio computacional propostos neste trabalho, foram usadas como referência as recomendações da norma IEEE Std 830 (IEEE, 1998) para a elaboração de um DR correto, não ambíguo, completo e consistente. Além disso, a estrutura do DR foi elaborada com base nos elementos da estrutura proposta pela norma, a princípio, foram usados os elementos: Visão Geral e Requisitos Específicos.

Neste capítulo é apresentada a fundamentação teórica para o desenvolvimento deste trabalho. Na Seção 2.2 são apresentadas as considerações sobre os requisitos de software. Na Seção 2.3 são descritos os conceitos de padrões de requisitos. Na Seção 2.4 são apresentados os trabalhos relacionados. Na Seção 2.5 é apresentada uma introdução aos sistemas de informação. Na Seção 2.6 são apresentados os fundamentos da norma IEEE Std 830 para a elaboração de um DR. As considerações finais são apresentadas na Seção 2.7.

2.2 Requisitos de Software

Um requisito é "uma condição ou capacidade necessária para um usuário resolver um problema ou alcançar um objetivo" (IEEE, 1990). Os requisitos de software são frequentemente classificados em funcionais - descrevem as funções que um sistema de software deve oferecer, não funcionais - descrevem as restrições sobre as funções do sistema e de domínio - descrevem as características e restrições do domínio do sistema, também conhecidos como regras de negócio.

A fase para elicitar, detalhar, documentar e validar os requisitos de um produto de software é denominada como Engenharia de Requisitos (ER). Se nessa fase os requisitos não forem devidamente identificados, estiverem incorretos, incompletos ou conflitantes, eles podem levar a um projeto fracassado.

O resultado principal da ER é um documento de Especificação de Requisitos de Software (ERS), que corresponde à descrição de todos os requisitos de um projeto. Um documento de especificação com qualidade contém requisitos especificados de forma clara, completa, sem ambiguidade, implementável, consistente e testável (PÁDUA, 2003).

O processo de ER definido por Sommerville (2007) é composto por quatro subprocessos, representados por elipses e os artefatos produzidos por esses, representados por retângulos, na Figura 2.1.

O Estudo de Viabilidade resulta em um relatório evidenciando a contribuição do sistema de software para os objetivos gerais da organização; sua implementação considerando as restrições definidas de custo e prazo e a tecnologia atual.

A Elicitação e Análise de Requisitos resulta geralmente em um documento de requisitos escrito em linguagem natural contendo todos os requisitos do cliente. Contudo, modelos que descrevem os processos de negócio, o problema a ser resolvido e o sistema a ser desenvolvido podem ser gerados para representar ou melhorar a compreensão dos requisitos. A *Unified Modeling Language* (UML) tornou-se a linguagem padrão para modelagem de sistemas orientados a objetos (BOOCH; RUMBAUGH; JACOBSON, 2006).

A Especificação de Requisitos resulta em uma descrição dos requisitos funcionais, não funcionais e domínio do usuário e do sistema.

A Validação de Requisitos corresponde à verificação de que os requisitos realmente expressam as necessidades reais do cliente, explicitados no Documento de Requisitos.

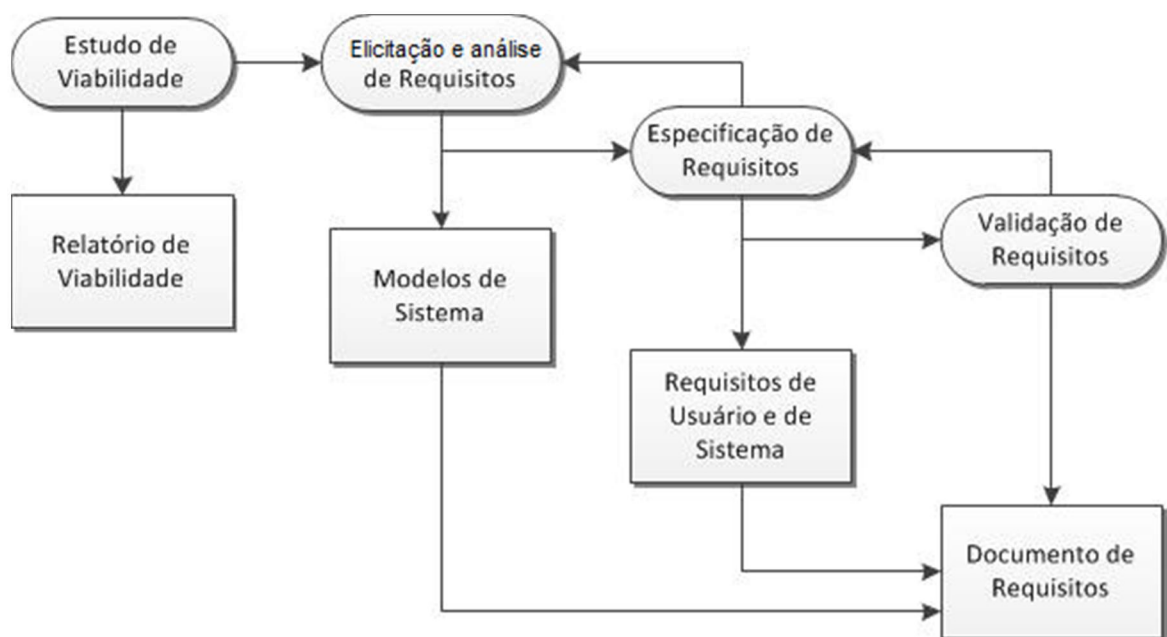


Figura 2.1 – Processo de Engenharia de Requisitos (SOMMERVILLE, 2007)

2.3 Padrões de Requisitos

O conceito de padrões foi apresentado na década de 70 pelo arquiteto Christopher Alexander, que identificou temas recorrentes na arquitetura para captar descrições e instruções de seus clientes (CHRISTOPHER, 2013). Na década de 90, os projetistas de software aplicaram a proposta de Alexander para o desenvolvimento de software, surgindo os padrões de projeto de software. Um padrão de projeto de software é descrito como uma solução recorrente para um problema em um determinado contexto que pode ser aplicada de modos diferentes (GAMMA et al., 1995) (SHALLOWAY; TROTT, 2004). Assim, padrões são utilizados em todas as fases do desenvolvimento de software, desde a análise até a implementação.

Na ER, padrões podem ser usados para criar uma linguagem comum para a disseminação de experiências sobre problemas recorrentes e suas soluções (WAHONO; CHENG, 2002).

Para Palomares et al. (2013), os requisitos não funcionais podem ser reusados independentemente do domínio, ou seja, aparecem da mesma forma em diferentes projetos. Ao contrário disso, a reutilização dos requisitos funcionais na maioria das vezes só é possível para um determinado domínio de software.

Withall (2007) define um catálogo com trinta e sete PRS organizados em oito domínios (Figura 2.2). Cada domínio possui padrões para atender às funcionalidades específicas desse domínio. No entanto, alguns padrões compartilham e integram informações, como ilustrado na Figura 2.2 por meio das setas que interligam alguns padrões.

Cada PRS foi especificado com a seguinte estrutura:

- **Detalhes básicos:** contém informações específicas relacionadas ao padrão, tais como: número da versão; data da última alteração; abordagem do requisito; padrões relacionados, se existir; a frequência prevista de utilização; sua classificação e a descrição do autor do padrão.
- **Aplicabilidade:** descreve as situações em que um padrão de requisito pode ser aplicado.
- **Discussão:** define as considerações para a escrita de um tipo de requisito.

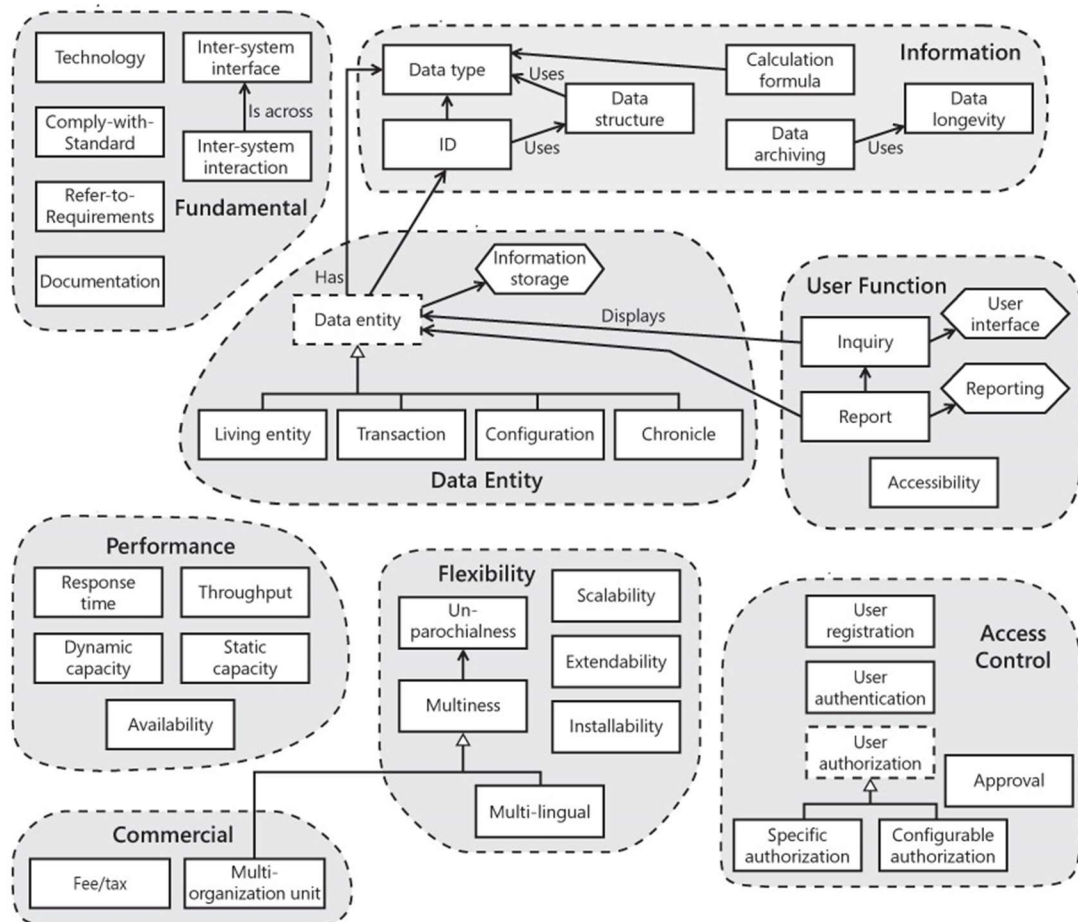


Figura 2.2 – Catálogo de Padrões de Requisitos de Software (WITHALL, 2007)

- **Conteúdo:** corresponde à informação que um requisito deve transmitir; pode descrever informações opcionais. Esta seção refere-se à essência do padrão.
- **Template(s):** usado como ponto de partida para escrever um ou mais requisitos; pode promover formas alternativas distintas.
- **Exemplo(s):** corresponde a um ou mais requisitos que foram escritos usando este padrão.
- **Requisitos extras:** correspondem à especificação dos requisitos adicionais que muitas vezes surgem na sequência de um requisito principal; declara que tipo de requisitos extras devem ser considerados e em que circunstâncias.
- **Considerações sobre o desenvolvimento:** oferece dicas para os projetistas e engenheiros de software sobre como implementar um requisito; deve ser escrito na linguagem de desenvolvedores.

- **Considerações sobre o teste:** oferece informações aos testadores do software sobre como testar um requisito; descreve principalmente como os testes de aceitação do usuário podem ser realizados.

Esta estrutura deve ser considerada apenas como uma indicação geral para descrever um padrão, no entanto não é a única abordagem possível para implementar uma abordagem baseada em PRS.

2.4 Trabalhos Relacionados

Roher e Richardson (2013) sugerem a incorporação de padrões de requisitos de sustentabilidade no processo de ER, a fim de facilitar a escrita do DR para determinados tipos de requisitos de sustentabilidade.

Wei, Xiaohong e Xuefei (2013) fornecem um *template* para a escrita de padrões de requisitos de segurança de software, elaborado com base nos padrões de projeto de Gamma et al. (1995) e os padrões de requisitos de Withall (2007). O padrão “*hazard control command*” por exemplo, é aplicável na especificação de requisitos de segurança crítica nos níveis de A-C, sendo que esses níveis correspondem ao grau de severidade do risco, sendo o nível A – catastrófico, B – crítico e C – significativo.

O estudo de Li et al. (2012) consiste na apresentação de dois padrões de requisitos de projetos típicos para sismologia computacional: o padrão de simulação e o padrão de acesso a dados. Os autores relatam que esses padrões podem beneficiar a especificação de projeto futuros e na redução de custos de desenvolvimento.

Franch et al. (2013) apresentam o *framework* PABRE (*PATtern-Based Requirements Elicitation*), que foi projetado para apoiar a reutilização de requisitos por meio de padrões. Esse *framework* consiste de um metamodelo que descreve os principais conceitos usando a noção de padrão, um método para conduzir os processos de elicitação e documentação de requisitos, um catálogo de padrões e uma ferramenta de apoio à gestão e utilização do catálogo. Essa ferramenta não está disponível e assim não foi possível a comparação dela com o apoio computacional desenvolvido neste trabalho.

O estudo de Palomares et al. (2012) apresenta dois padrões requisitos não técnicos elaborados por meio do *framework* PABRE. Além disso, afirmam que esses padrões integram um catálogo maior, que atualmente contém 37 padrões de requisitos não técnicos, 29 padrões de requisitos não funcionais e 47 padrões de requisitos funcionais que se aplicam ao domínio de gestão de documentos e que foram construídos a partir de documentos de requisitos de projetos anteriores.

Os padrões encontrados nesses trabalhos relacionados, em sua maioria, são para a especificação de requisitos não funcionais, que propicia reuso do padrão na íntegra em diferentes domínios, sem qualquer alteração. Para os requisitos funcionais isso não ocorre, uma vez que há necessidade de especificar detalhes para o sistema em desenvolvimento (PALOMARES et al., 2013). Contudo, observa-se que no domínio de SI existe similaridade na especificação dos requisitos funcionais independentemente do tipo de sistema. Essa similaridade pode ser encontrada por exemplo, na manutenção de dados (CRUD), nos relatórios gerenciais, nas transações comerciais, entre outras funções.

Um exemplo de um padrão de requisito não funcional de sustentabilidade que pode ser aplicado em diferentes domínios é apresentado no Quadro 2.1.

Este trabalho difere dos demais, pois:

- a) permite a elaboração padronizada de um DR com requisitos funcionais e regras de negócio instanciados por meio de padrões de requisitos para sistemas específicos no domínio de SI. Por exemplo, o padrão que especifica o cadastro de um novo registro de uma entidade, pode ser usado para a elaboração de um sistema hotel, um sistema de farmácia ou qualquer outro e terá no DR a especificação completa desse requisito;
- b) usa um apoio computacional para elaborar o DR;
- c) o engenheiro de software pode reusar especificações de requisitos construídas em sistemas anteriores modificando alguns atributos ou não. Por exemplo, uma especificação para a reserva em um hotel, pode ser recuperada do repositório e reusada em um outro sistema de hotel. Há a possibilidade de reusar a função de reserva para um sistema que a realize de forma similar, por exemplo, um sistema de reserva de livros, carros, produtos, etc.;
- d) novos padrões de requisitos podem ser elaborados usando a estrutura de apresentação.

Quadro 2.1 – Padrão: Communicating Resource Consumption (ROHER; RICHARDSON, 2013)**Summary**

Pattern Domain:

This pattern can be used for various types of resources within several different domains such as, packaging in e-commerce systems, personal home resource tracking systems, and so on.

Frequency:

This pattern can be used to create an unlimited number of requirements.

Author(s):

Kristin Roher

Applicability

This pattern should be used to specify a way in which a system should communicate resource consumption to a user. This may include resource consumption of the system itself or resource consumption affected by the use of the system. This pattern should not be used to incentivize a user to reduce resource consumption. Although communication can be a form of incentive, the “incentivizing minimal resource consumption” pattern should be used instead.

This pattern is meant to address the problem of users being unaware of resource consumption. By making users aware of this, they begin to understand the impact they or the system they are using are having on the environment in a tangible way. This allows them to then adjust their consumption and consider new ways to perform tasks that use less of a particular resource based on patterns they see in the observed consumption. For example, if the user can see that they save X amount of electricity when they power off their laptop while sleeping, they will understand the direct impact they can have on the environment. To encourage the user to continue a task that reduces their resources consumption, please refer to the “incentivizing minimal resource consumption” pattern.

Content

Requirements derived from this pattern may contain the following:

1. The resource to be communicated. For example, energy/electricity, packaging materials, water, gas, etc.
2. Motivation behind why that requirement is important to the system, the user, or the environment.
3. In what way the system shall communicate the amount of consumption. This may be through charts and graphs, raw numbers and tables, percentages, comparison of other users, etc.
4. When this information should be displayed. The following questions should be considered: Should it be displayed as a notification when a certain amount of a resource is consumed? Is it an option the user has to select to be able to see or is it always on display for the user to see?

Archetype

The following is a generalized requirement from which more detailed instances can be derived and tailored:

“The system shall communicate information to the user about the amount of each resource consumed by the system or resource consumption affected by use of the system.” **Examples**

- The system shall provide a notification to the user when their resource consumption is approaching the maximum benchmark for that resource.
- The system shall display the average and minimum amounts for each type of resource consumed by other users.
- The system shall display the amount of resource consumption in a way that the user will understand.
- The system shall provide options for ways in which the information is displayed.

Discussion

To properly use this pattern, a way to measure consumption within the system will have to be established. We are currently working on developing widgets that will do most of this work for you and the pattern will be updated once they are completed. A set of benchmarks may also need to be created to decide when a user is being wasteful/conservative in their amount of consumption. Some sample benchmarks for this are also in progress.

Related Patterns

- Incentivizing Minimal Resource Consumption
- Tracking Resource Consumption

2.5 Conceitos em Sistemas de Informação

O processamento de informação comercial corresponde à maior área de aplicação de software. Diferentes sistemas, como por exemplo de contas a pagar e receber, vendas, entre outros evoluíram para softwares de SI que fornecem acesso a banco de dados com informações comerciais. Esses SI facilitam as operações comerciais e as tomadas de decisões administrativas (REZENDE, 2005).

Em razão da existência de diferentes interesses, especialidades e níveis em uma organização, os sistemas são classificados em tipos, sendo os mais comuns: sistemas de processamento de transações (SPT), sistemas de informação gerencial (SIG) e sistemas de apoio a decisões (SAD) (STAIR; REYNOLDS, 2006). Os SPTs são usados para registrar transações básicas da organização, tais como registro de pedidos de venda, reservas de hospedagem em hotel, vendas, manutenção do registro do cliente etc. Os SIGs usam os dados armazenados pelos SPTs para fornecer informações para gerentes e tomadores de decisões, como exemplo pode-se citar um relatório de pedidos semanal e a lista de reservas do hotel de um determinado período. Os SADs são usados para dar apoio à tomada de decisões em problemas específicos, contribuindo para responder questões como: Qual o impacto na programação de produção caso ocorra duplicação nas vendas em dezembro? Os SADs usam informações internas dos SPTs, SIGs e frequentemente recorrem a informações de fontes externas (LAUDON; LAUDON, 2007).

Os requisitos funcionais desses tipos de sistemas, na maioria dos casos, requerem a definição de regras de negócio específicas. Na perspectiva de sistemas de informação, uma regra de negócio é uma declaração que define ou restringe alguma circunstância do negócio (WIEGERS; BEATTY, 2013). Uma regra de negócio está sempre relacionada com algum requisito funcional.

Wiegiers e Beatty (2013) identificam cinco tipos principais de esquemas de classificação de regras de negócio: fatos, restrições, ativadores de ações, inferências e computações. Cada um deles apresenta uma forma típica de escrita, como apresentado a seguir:

- **Fatos:** são declarações significativas sobre a forma de execução do negócio em questão. Um fato descreve associações ou relacionamentos entre

importantes termos do negócio. Exemplo: Todo pedido tem uma taxa de entrega.

- **Restrições:** são declarações que restringem as ações que o sistema ou seus usuários podem realizar. Exemplo: Um estudante só pode tomar emprestado, concomitantemente, de um a três livros.
- **Ativadores de Ações:** são regras que disparam alguma ação se condições específicas forem verdadeiras. Exemplo: Se a retirada do livro não ocorrer dentro do prazo determinado, então a reserva é cancelada.
- **Inferências:** são regras que derivam novos fatos a partir de outros fatos. Geralmente são escritas no padrão “se / então”, no entanto, a cláusula “então” simplesmente fornece um fato ou conhecimento e não uma ação a ser tomada. Exemplo: Se o usuário não devolver um livro emprestado dentro do prazo estabelecido, então ele torna-se um usuário pendente.
- **Computações:** são regras de negócio que definem cálculos que transformam dados existentes em novos dados usando algoritmos ou fórmulas matemáticas específicas. Podem ser expressas como fórmulas matemáticas, descrição textual, tabelas, etc. Exemplo: Aplica-se um desconto progressivo se mais do que 10 unidades forem adquiridas. De 6 a 10 unidades, o desconto é de 10%. A compra de 11 ou mais unidades tem um desconto de 20%.

Esses conceitos foram utilizados neste trabalho, entretanto na definição dos padrões de regra de negócio, não foi usado esse esquema de classificação com o intuito de facilitar a utilização. Contudo, o engenheiro de software deve conhecê-los de forma a elaborar um documento de requisitos completo e sem ambiguidade, observando a especificação dos requisitos funcionais, não funcionais e das regras de negócio de um sistema de software.

Este trabalho contempla padrões para a escrita de requisitos funcionais e de regras de negócio. A elaboração de padrões para requisitos não funcionais não faz parte do escopo deste trabalho.

2.5.1 Reúso no Desenvolvimento de Sistemas de Informação

Atualmente, as empresas operam em um ambiente sujeito a rápidas mudanças. Nesse cenário, precisam responder a novas oportunidades e mercados, mudanças econômicas e ao surgimento de produtos e serviços concorrentes. Conseqüentemente, é essencial que um novo SI seja desenvolvido rapidamente para aproveitar essas oportunidades e às pressões da concorrência (SOMMERVILLE, 2007).

Nesse sentido, a reusabilidade no desenvolvimento e manutenção de software ajuda os engenheiros de software na redução do tempo de trabalho e na garantia da qualidade do novo SI. O reúso acontece por meio da utilização de partes do software já avaliadas e/ou prontas em novos projetos, e pode acontecer em todos os níveis do projeto, desde a análise até a implementação.

Braga, Germano e Masiero (1999) propuseram reúso para o desenvolvimento de SI de Gestão de Recursos de Negócios (GRN), com a utilização de uma linguagem de padrões de análise. A notação de diagrama de classes da UML é utilizada para expressar cada um dos padrões, o que colabora com a especificação e posterior implementação de um sistema de software desse domínio. Na Figura 2.3 é apresentado o grafo que ilustra os 15 (quinze) padrões da linguagem de padrões GRN, bem como as dependências entre eles e a ordem na qual são geralmente aplicados.

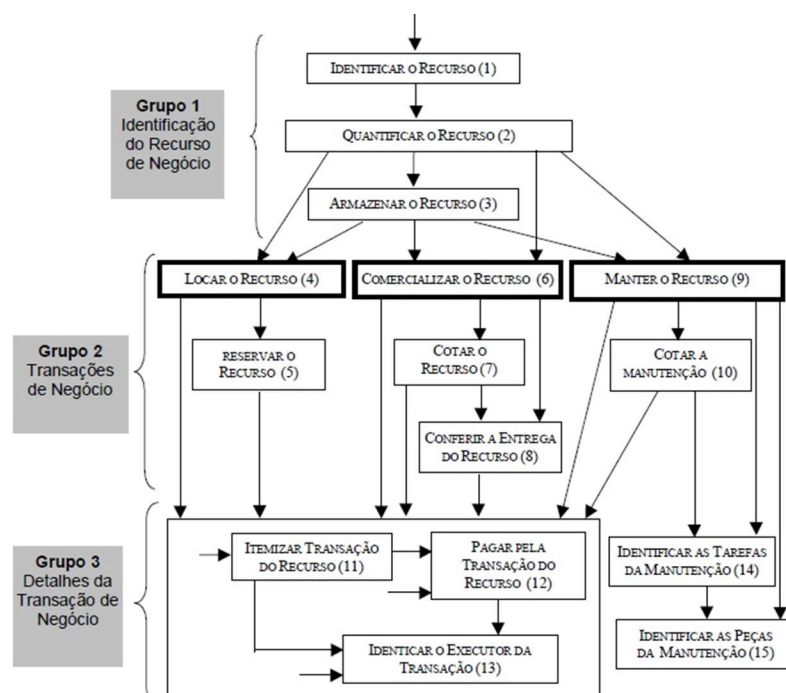


Figura 2.3 – Linguagem de Padrões para Gestão de Recursos de Negócios (BRAGA; GERMANO; MASIERO, 1999)

Para exemplificar, na Figura 2.4 é mostrado o diagrama de classes do padrão “Identificar o Recurso”. Esse padrão pode ser aplicado em sistemas que lidam com uma ou mais das seguintes transações: pedidos, compras, vendas, locações, alugueis, reservas, etc. Essas transações referem-se por exemplo, a produtos de uma loja, fitas de vídeo em uma locadora, livros em uma biblioteca, carros em uma oficina mecânica. Todos esses casos tratam de recursos de negócio gerenciados por sistemas específicos (BRAGA; GERMANO; MASIERO, 1999).



Figura 2.4 – Padrão Identificar o Recurso (BRAGA; GERMANO; MASIERO, 1999)

Na Figura 2.5 está ilustrado um exemplo de instanciação do padrão “Identificar o Recurso” para um sistema de locadora de vídeos, em específico para o recurso “Vídeo”.

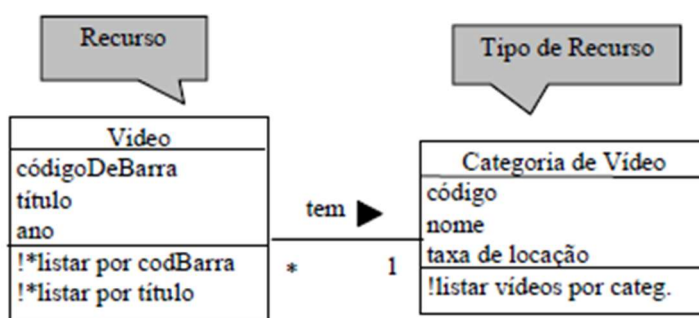


Figura 2.5 – Instanciação do Padrão Identificar o Recurso (BRAGA; GERMANO; MASIERO, 1999)

Para cada um dos padrões, Braga, Germano e Masiero (1999) apresentam a solução por meio de um diagrama de classes e um exemplo de aplicação conforme é ilustrado na Figura 2.5. Esse trabalho fornece uma importante contribuição aos engenheiros de software, principalmente os menos experientes durante a fase de análise de requisitos de projetos de GRN.

2.6 A norma IEEE Std 830

A norma IEEE Std 830 (IEEE, 1998) descreve recomendações para a especificação de requisitos de software (ERS) para que seja elaborado um DR de software com qualidade. As características indicadas pela norma são apresentadas a seguir:

- **Correto:** considera-se uma ERS correta, aquela que expressa todos os requisitos que o software deve atender.
- **Não ambíguo:** cada requisito declarado deve ter somente uma interpretação.
- **Completo:** precisa conter todos os requisitos funcionais e não funcionais, os atributos, respostas às entradas, bem como legendas e referências à figuras, tabelas, diagramas e definição de termos.
- **Consistente:** nenhum subconjunto individual de requisitos descrito deve entrar em conflito.
- **Classificado por importância e/ou estabilidade:** cada requisito deve ser classificado por importância e/ou estabilidade. Por exemplo, um requisito pode ser classificado como essencial ou desejável.
- **Verificável:** um requisito é verificável, se existe um processo por meio do qual uma pessoa ou uma máquina possa verificar que o software cumpre esse requisito.
- **Modificável:** as mudanças em requisitos devem ser efetuadas de forma fácil, completa e consistente.
- **Rastreável:** facilita a identificação do requisito em versões futuras do desenvolvimento ou na documentação.

Para cumprir essas características, a norma fornece uma estrutura (Quadro 2.2) para a elaboração do DR. Entretanto, nem todos os tópicos da estrutura são necessários para todos os documentos de requisitos. Cada organização deve adaptar o padrão de acordo com o tipo de sistema que desenvolve.

Quadro 2.2 – Conteúdo do Documento de Especificação de Requisitos (IEEE, 1998)**Tabela de conteúdos****1 Introdução****1.1 Objetivo**

Delimita o objetivo e especifica o público alvo da ERS.

1.2 Escopo

Define o que o software fará e se necessário o que não fará. Inclui a descrição da aplicação, os benefícios relevantes, os objetivos e as metas.

1.3 Definições, siglas e abreviações

Fornecer as definições de todos os termos, siglas e abreviações necessárias para interpretar a ERS.

1.4 Referências

Fornecer uma lista de todos os documentos referenciados na ERS, bem como as origens das referências.

1.5 Visão Geral

Descreve como a ERS está organizada.

2 Descrição geral**2.1 Perspectiva do produto**

Coloca o produto em perspectiva com outros produtos relacionados. Se o software é independente, seu conteúdo próprio é descrito e se o software é um componente de um grande sistema, deve ser relatado os requisitos desse sistema para a funcionalidade do software e as interfaces entre o sistema e o software.

2.2 Funções do produto

Fornecer um sumário das funções do software, sem mencionar os detalhes.

2.3 Características do usuário

Descreve as características gerais dos usuários do software incluindo o nível educacional, experiência e habilidade técnica.

2.4 Restrições

Fornecer uma descrição geral de qualquer item que limite as opções do desenvolvedor. Como por exemplo: políticas reguladoras e limitações de hardware.

2.5 Pressupostos e dependências

Lista dos fatores que afetam os requisitos expressos no documento de ERS.

3. Requisitos Específicos

Contém todos os requisitos do software num nível de detalhe suficiente para que os desenvolvedores estejam aptos para satisfazer estes requisitos no desenvolvimento do sistema e testarem para que o sistema satisfaça estes requisitos. Os requisitos expressos devem ser compreendidos pelos usuários e operadores.

Apêndices

Os apêndices nem sempre são considerados parte da especificação de requisitos e nem sempre são necessários. Eles podem incluir: uma descrição dos problemas a serem resolvidos pelo software, pacotes de instruções especiais para o código etc.

Índice

Os índices são importantes e seguem as práticas gerais.

A abordagem de reuso para a elaboração do DR seguiu algumas das características recomendadas pela norma do IEEE, por meio das seguintes diretrizes:

- **Correto:** possuir um conjunto suficiente de padrões, para o domínio de SI que oriente o engenheiro de software na completa especificação de DRs;
- **Não ambíguo e Completo:** realizar completa análise nos requisitos utilizados, como fonte de conhecimento para a elaboração dos padrões, que haja apenas uma interpretação e que expresse todas as informações pertinentes ao problema;
- **Consistente:** expressar a solução de um único problema, em cada padrão;
- **Rastreável:** possibilitar o estabelecimento de relação entre os requisitos e que cada requisito do DR possua uma identificação única.

Outras características estão fora do escopo deste trabalho e estão previstas em trabalhos futuros, como a gestão de modificação do DR, a elaboração dos padrões de requisitos não funcionais, a classificação dos requisitos por importância e a verificação de requisitos. Quanto ao conteúdo do documento proposto pela norma, serão usadas apenas as seções de Visão Geral e Requisitos Específicos.

2.7 Considerações Finais

O reúso de requisitos com base em padrões é uma solução que pode amenizar os problemas da especificação de requisitos. Pode também propiciar benefícios como a economia de tempo, pois cada requisito não precisa ser escrito a partir do zero, uma vez que são reusadas soluções comprovadas anteriormente (PALOMARES et al., 2013) (WITHALL, 2007). Aproveitar os conhecimentos adquiridos em projetos anteriores é uma estratégia adequada para melhorar a qualidade dos requisitos e em consequência aumentar as chances de sucesso de novos projetos, assim como aumentar a eficiência do processo de levantamento de requisitos (FRANCH et al., 2013).

Os SI correspondem a maior área de aplicação de software. Esses sistemas facilitam as operações comerciais e as tomadas de decisões. Devido à alta competitividade e as rápidas mudanças no cenário empresarial, os SI passaram a receber maiores investimentos e atenção pelas empresas.

Nos trabalhos relacionados, os padrões encontrados, em sua maioria, são para a escrita de requisitos não funcionais, o que propicia reúso do padrão na íntegra em

diferentes domínios, sem qualquer alteração. Para os requisitos funcionais isso não ocorre, uma vez que há necessidade de especificar detalhes para o sistema em desenvolvimento (PALOMARES et al., 2013). Contudo, observa-se que no domínio de SI existe similaridade na escrita dos requisitos funcionais independentemente do tipo de sistema, o que indica um possível padrão de requisito.

A norma IEEE Std 830 fornece um conjunto de características que deve ser seguida ou adaptada pelo engenheiro de software para a elaboração de um DR completo e consistente. Os padrões elaborados e o apoio computacional desenvolvido neste trabalho seguiram algumas das recomendações propostas pela norma para a escrita de requisitos.

No próximo Capítulo é apresentada a metodologia usada para a elaboração dos padrões de requisitos, bem como os padrões elaborados.

Capítulo 3

PADRÕES DE REQUISITOS FUNCIONAIS PARA SISTEMAS DE INFORMAÇÃO

3.1 Considerações Iniciais

Neste capítulo são apresentados padrões de requisitos que podem ser usados em diversos domínios, pois permitem a identificação e escrita de requisitos para as quatro principais operações encontradas em SI, sendo: as operações básicas de manipulação de dados - CRUD (acrônimo de *Create*, *Read*, *Update* e *Delete* na língua inglesa). Também são apresentados os padrões elaborados neste trabalho para o domínio de SI, referentes ao processamento de transações, à consulta/impressão de informações gerenciais e a algumas regras de negócio comumente existentes em software do domínio de SI.

Na escrita de requisitos funcionais, geralmente o texto possui uma parte fixa e outra variável. A parte fixa representa a operação a ser realizada e a parte variável contém os detalhes específicos dessa operação, como: a entidade de dados atendida, os atributos necessários, restrições específicas do negócio, entre outros. Os padrões de requisitos de SI elaborados neste trabalho contemplam esse formato. A utilização desses padrões por meio de um apoio computacional é apresentada no Capítulo 4.

Na Seção 3.2 é apresentada a metodologia utilizada para a elaboração dos padrões de requisitos funcionais para o domínio de SI. Na Seção 3.3 são descritos os padrões de requisitos funcionais e de regras de negócio elaborados. As considerações finais são apresentadas na Seção 3.4.

3.2 Metodologia de Elaboração

Foi estabelecida uma metodologia para a obtenção de conhecimento, que represente a função atendida por cada padrão, a partir de DR disponíveis na literatura e academias. A metodologia consistiu das seguintes etapas:

1. **Obtenção de diferentes documentos de requisitos:** foram obtidos documentos de requisitos fictícios utilizados na academia, como material didático de disciplinas, em instituições de ensino (UFSCar, USP e UEMG) e outros que estão disponibilizados na literatura especializada, bem como na web;
2. **Organização dos Requisitos:** os requisitos de diferentes documentos do domínio de SI foram organizados por tipo de operação. Exemplos: manutenção de dados (criar, ler, atualizar e excluir), processamento de transação (compra, venda, retirada, devolução, etc.), impressão ou listagem de relatórios gerenciais, etc. Os requisitos que realizam a mesma operação foram os selecionados a se tornarem um padrão;
3. **Análise dos Requisitos:** para cada tipo de operação, os requisitos foram analisados para identificar se estavam completos e se não possuíam ambiguidade, bem como para identificar as partes do texto que deveriam ser fixas e as partes variáveis;
4. **Verificação da Existência de um Padrão:** para atender ao conjunto de requisitos analisados, uma pesquisa na literatura deve ser realizada a fim de identificar se já existe algum padrão que atenda ao problema identificado. Caso não exista, passar para a etapa 5, senão, passar para a etapa 6.
5. **Especificação do Padrão:** foi definida a aplicabilidade e a solução do padrão para atender a escrita de um conjunto de requisitos similares (Quadro 3.1). Além disso, para apoiar o engenheiro de software na utilização do padrão, para cada parâmetro (parte variável do requisito) do *template* foram estabelecidas sugestões de valores e a multiplicidade para orientar a quantidade de valores de preenchimento (ex. a multiplicidade 1..* indica que o parâmetro pode possuir um ou muitos valores);

6. **Estabelecimento de Relação entre os Padrões:** definição da relação do padrão em questão com outros padrões. Essa relação pode orientar e/ou alertar o engenheiro de software na utilização de outros padrões que complementem a especificação, quando da aplicação de um determinado padrão, proporcionando completude no DR.

A metodologia adotada para a elaboração dos padrões (Figura 3.1) seguiu uma abordagem iterativa e incremental. A cada ciclo um padrão que atende a um problema específico é elaborado e a relação do mesmo com os demais já elaborados é identificada. Para a identificação da relação entre os padrões, uma nova análise nos requisitos já organizados e nos DRs são realizadas.

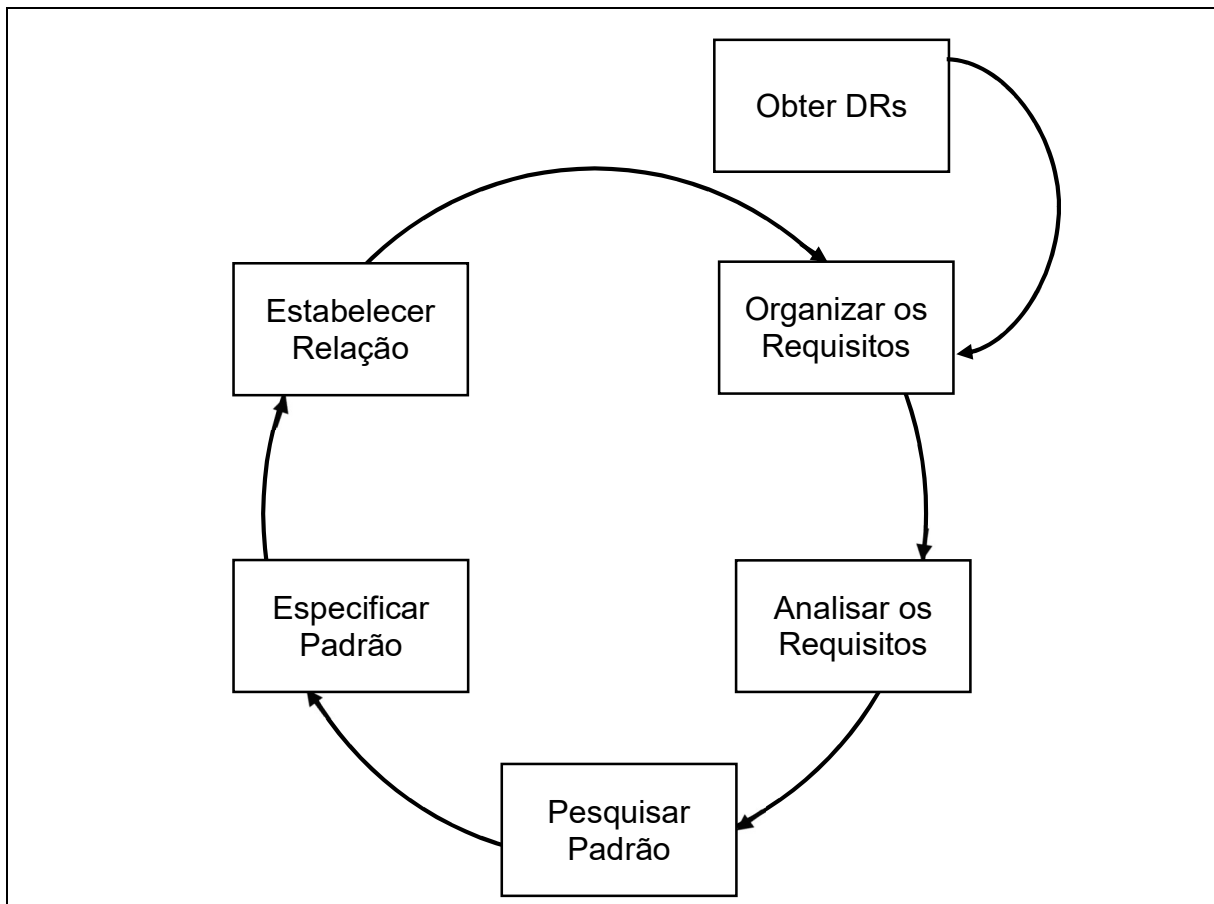


Figura 3.1 – Metodologia para Elaboração dos Padrões

Nesta dissertação foram utilizados três documentos de requisitos para a elaboração dos padrões: de hotel, de biblioteca e de locação de carros. A seguir são apresentadas amostras dos requisitos que contribuíram para a elaboração dos padrões.

- 1) Requisitos de manutenção de dados:
 - a) Requisitos de Software para Locação de Carros:

“O sistema deve permitir a inclusão dos carros para aluguel pertencentes à locadora de carros. Cada carro possui os seguintes atributos: placa do carro, código da categoria do carro, fabricante, marca, ano e modelo do carro”.

b) Requisitos de Software para Biblioteca:

“O sistema deve permitir a inclusão de leitores da biblioteca, com os seguintes atributos: nome, endereço, cidade, estado, telefone, e-mail, documento de identificação, categoria de leitor e data de nascimento”.

c) Requisitos de Software para Hotel:

“O sistema deve permitir a inclusão dos tipos de acomodação oferecidos pelo hotel, com os seguintes atributos: código do tipo de acomodação, descrição do tipo de acomodação, quantidade de unidades total desse tipo de acomodação existente no hotel, preço da diária, número de pessoas adultas e número de crianças que esse tipo de acomodação comporta”.

2) Requisitos de processamento de transações:

a) Requisitos de Software para Locação de Carros:

“O sistema deve permitir o processamento da reserva de carro, com os seguintes atributos: data e hora de retirada, data e hora previstas para devolução, identificação do cliente, categoria de carro desejada, valor da diária e funcionário responsável”.

b) Requisitos de Software para Biblioteca:

“O sistema deve permitir o empréstimo de uma obra literária por um leitor. Cada empréstimo possui os seguintes atributos: data de empréstimo, data prevista para devolução, identificação do leitor, funcionário responsável pelo empréstimo e identificação da cópia da obra emprestada

c) Requisitos de Software para Hotel:

“O sistema deve permitir a reserva de acomodação. Cada reserva possui os seguintes atributos: data e hora de chegada do hóspede, data e hora de saída do hóspede, identificação do hóspede principal, tipo de acomodação desejada, nomes e idades dos acompanhantes e valor da diária”.

3) Requisitos de regras de negócio:

a) Requisitos de Software para Locação de Carros:

“A remoção de reserva somente é permitida sem maiores encargos até 4 horas antes do início previsto para aluguel do carro”.

b) Requisitos de Software para Biblioteca:

“A reserva de obras somente deve ser permitida se houverem obras disponíveis no período indicado”.

c) Requisitos de Software para Hotel:

“A reserva de acomodação somente deve ser concretizada se houver vagas suficientes para atendê-la”.

Para atender ao domínio de SI, neste trabalho, foi estabelecida uma estrutura de apresentação para os padrões apresentada no Quadro 3.1.

Quadro 3.1 – Estrutura Adotada para Apresentação dos Padrões

Elemento do Padrão	Descrição
Nome	Especifica o nome do padrão, que deve ser único e refletir a aplicabilidade do padrão.
Domínio	Corresponde ao domínio de aplicação do padrão.
Propósito	Descreve o objetivo da aplicação do padrão.
Problema	Descreve a situação em que o padrão pode ser aplicado.
Consequência	Descreve as consequências de se utilizar o padrão.
Tipo	Especifica o tipo de requisito atendido pelo padrão: Funcional, não funcional ou regra de negócio.
Solução	Apresenta um <i>template</i> para a escrita da parte fixa e variável do requisito que o padrão deve representar. A parte fixa apresenta um texto padrão para a escrita do requisito. A notação <...> é usada para descrever a parte variável que é denominada de parâmetro e deve ser substituída pelos dados pertinentes ao requisito a ser elaborado.
Padrões Relacionados	Especifica os padrões relacionados que são complementares ao padrão em questão. Esse elemento contribui para o engenheiro de software na indicação de outros possíveis padrões que podem ser usados quando ocorrer o uso do padrão em questão.

Os elementos *nome*, *propósito*, *problema*, *consequência* e *solução* da apresentação dos padrões foram reusados da proposta por Gamma et al. (1995). O elemento domínio foi incluído para permitir a definição do domínio de aplicação do padrão, uma vez que a reutilização dos requisitos funcionais geralmente é restrita a um único domínio. O elemento tipo fez-se necessário para estabelecer o tipo de requisito atendido pelo padrão; e o elemento padrões relacionados foi reusado da proposta de Withall (2007), afim de contribuir para a completa especificação do DR.

3.3 Padrões de Requisitos Elaborados

Para atender ao domínio de SI, foi elaborado um catálogo com onze padrões de requisitos funcionais, divididos em quatro grupos de operações, sendo:

a) operações básicas: representam os padrões que atendem as funções de manipulação de dados (CRUD). As operações de manipulação de dados também são comuns em outros domínios. Nesse sentido, os padrões elaborados para esse contexto, podem ser reusados na especificação de requisitos de outros domínios;

b) processamento de transação: representa o padrão para a realização e o armazenamento de informação de transações rotineiras;

c) informações gerenciais: representa o padrão para a realização de consultas de informações gerenciais; e

d) regras de negócio: representam os padrões que estabelecem as condições de operações do sistema.

O catálogo de padrões é apresentado no Quadro 3.2. O tipo de operação está na primeira coluna, o nome do padrão na segunda e a consequência é exibida na terceira coluna. Nas próximas subseções os padrões são apresentados em detalhes.

Quadro 3.2 – Catálogo de Padrões de Requisitos para o Domínio de Sistemas de Informação

Tipos de Operações	Nome do Padrão	Consequência
Operações Básicas	Incluir Informação	A escrita de um requisito que descreve a necessidade de um usuário de armazenar informação de uma entidade de dados.
	Alterar Informação	A escrita de um requisito que descreve a necessidade de um usuário de alterar determinados dados de uma entidade de dados.
	Excluir Informação	A escrita de um requisito que descreve a necessidade de um usuário de excluir informação de uma entidade de dados.
	Manter Informação	A escrita de um requisito que descreve a necessidade de um usuário de incluir, alterar e excluir informação de uma entidade de dados.
	Recuperar Informação	A escrita de um requisito que descreve a necessidade de um usuário de recuperar informação de uma entidade de dados por meio da descrição de parâmetros.
Processamento de Transações	Processar Transação	A escrita de um requisito que descreve a necessidade de um usuário de processar uma transação e armazenar sua informação.
Informações Gerenciais	Especificar Consulta Gerencial	A escrita de um requisito que descreve a necessidade de um usuário de realizar consultas de informações gerenciais em tela ou impresso.
Regras de Negócio	Informação Requerida	A escrita de um requisito que descreve uma regra de negócio que estabelece as informações requeridas por uma determinada operação.
	Condição de Execução	A escrita de um requisito que descreve uma regra de negócio que estabelece as condições de execução de uma determinada operação.
	Límite de Execução	A escrita de um requisito que descreve uma regra de negócio que estabelece limites específicos de execução de uma determinada operação.
	Cálculo de Valor	A escrita de um requisito que descreve a necessidade de um usuário de especificar um determinado cálculo matemático, requerido em uma operação.

3.3.1 Padrões de Requisitos: Operações Básicas

Os padrões de requisitos para as operações básicas são aplicáveis para a escrita de requisitos que descrevem a manipulação de dados, tais como criar, ler, atualizar e excluir informação. Cada padrão atende a uma dessas operações. No Quadro 3.3 é apresentado o padrão Incluir Informação, no Quadro 3.4 o padrão Alterar Informação, no Quadro 3.5 o padrão Excluir Informação, no Quadro 3.6 o padrão Recuperar Informação e no Quadro 3.7 o padrão Manter Informação que reúne em um único padrão as operações de criar, atualizar e excluir informação. Após a análise de DRs de diferentes sistemas, percebeu-se que as operações de manipulação de dados, apesar de serem distintas, para sistemas simples é comum encontrar essas operações em um único requisito. Nesse sentido o padrão Manter Informação foi elaborado para descrever em um único requisito as operações de criar, atualizar e excluir informação.

Quadro 3.3 – Padrão Incluir Informação

Nome	Incluir Informação
Domínio	Sistemas de Informação
Propósito	Armazenar informações de uma entidade de dados.
Problema	Estabelecer os dados da entidade que devem ser armazenados.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de armazenar informação de uma entidade de dados.
Tipo	Funcional
Solução	Template: Permitir a inclusão de informações de <entidade>, contendo os seguintes atributos: <atributos>.
	Parâmetro: <entidade> nome da entidade que contém informações do domínio (1). <atributos> nome dos atributos da entidade que devem ser armazenados (1..*).
	Exemplo de Parâmetros: <entidade> cliente <atributos> código, nome, endereço, telefone....
	Exemplo da Descrição do Requisito: Permitir a inclusão de informações de cliente, contendo os seguintes atributos: código, nome, endereço, bairro e cidade.
Padrões Relacionados	Alterar Informação Excluir Informação Recuperar Informação

Quadro 3.4 – Padrão Alterar Informação

Nome	Alterar Informação
Domínio	Sistemas de Informação
Propósito	Alterar informações de uma entidade de dados.
Problema	Estabelecer quais dados da entidade podem ser alterados.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de alterar determinados dados de uma entidade de dados.
Tipo	Funcional
Solução	Template: Permitir a alteração de <entidade> no(s) seguinte(s) atributo(s): <atributos>.
	Parâmetro: <entidade> nome da entidade que contém informações do domínio (1). <atributos> nome dos atributos da entidade que podem ser alterados(1..*).
	Exemplo de Parâmetros: <atributos> código, nome, endereço <entidade> cliente, produto, vendedor
	Exemplo da Descrição do Requisito: Permitir a alteração de cliente no(s) seguinte(s) atributo(s): endereço. Permitir a alteração de cliente no(s) seguinte(s) atributo(s): endereço e bairro.
Padrões Relacionados	Incluir Informação

Quadro 3.5 – Padrão Excluir Informação

Nome	Excluir Informação
Domínio	Sistemas de Informação
Propósito	Excluir informações de uma entidade de dados.
Problema	Excluir a informação desejada.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de excluir informação de uma entidade de dados.
Tipo	Funcional
Solução	Template: Permitir a exclusão de um registro de <entidade><condição>.
	Parâmetro: <entidade> nome da entidade que contém informações do domínio (1). <condição> descrição da condição para a exclusão do registro (1..*).
	Exemplo de Parâmetros: <entidade> cliente, produto, vendedor <condição> desde que não possua nenhuma transação vinculado ao mesmo.
	Exemplo da Descrição do Requisito: Permitir a exclusão de um registro de cliente. Permitir a exclusão de um registro de cliente desde que não possua nenhuma transação vinculada a ele.
Padrões Relacionados	Incluir Informação

Quadro 3.6 – Padrão Recuperar Informação

Nome	Recuperar Informação
Domínio	Sistemas de Informação
Propósito	Recuperar informações de uma entidade de dados.
Problema	Recuperar uma informação.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de recuperar informação de uma entidade de dados por meio da descrição de parâmetros.
Tipo	Funcional
Solução	Template: Permitir a recuperação de informação de <entidade> pelo(s) atributo(s) <atributos>, <condição>.
	Parâmetro: <entidade> nome da entidade que contém informações do domínio (1). <atributos> nome dos atributos da entidade que podem receber parâmetro (1..*) <condição> descrição da condição para a recuperação da informação (0..*). Exemplo de Parâmetros: <entidade> cliente, produto, vendedor <atributos> código, nome, cpf <condição> sempre que o cliente for selecionado, durante o processamento da operação.
	Exemplo da Descrição do Requisito: Permitir a recuperação de informação de cliente pelo(s) atributo(s) nome e cpf. Permitir a recuperação de informação de reserva pelo(s) atributo(s) número da reserva, durante o processamento da operação de venda.
Padrões Relacionados	Incluir Informação

Quadro 3.7 – Padrão Manter Informação

Nome	Manter Informação
Domínio	Sistemas de Informação
Propósito	Permitir incluir, alterar e excluir informação de uma entidade de dados.
Problema	Estabelecer os dados da entidade.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de incluir, alterar e excluir informação de uma entidade de dados.
Tipo	Funcional
Solução	Template: Permitir a inclusão, alteração e exclusão de <entidade>, com os seguintes atributos: <atributos>.
	Parâmetro: <entidade> nome da entidade que contém informações do domínio (1). <atributos> nome dos atributos da entidade (1..*). Exemplo de Parâmetros: <entidade> cliente, produto, vendedor <atributos> código, nome, endereço
	Exemplo da Descrição do Requisito: Permitir a inclusão, alteração e exclusão de Cliente com os seguintes atributos: código, nome, endereço e telefone.
Padrões Relacionados	-

3.3.2 Padrão de Requisito: Processamento de Transação para SI

A maioria das organizações necessita de sistemas que realizem e registrem as transações rotineiras necessárias para o funcionamento da empresa, tais como pedidos de venda, de reservas de hotel, contas a pagar e receber, entre outras (LAUDON; LAUDON, 2007). Esse tipo de sistema, denominado Sistemas de Processamento de Transação (SPT), oferece suporte às atividades do dia-a-dia e ajudam as organizações a agregar valor a seus produtos e serviços (STAIR; REYNOLDS, 2006).

Para contribuir com a identificação e escrita de requisitos relacionados aos SPTs, o padrão Processar Transação apresentado no Quadro 3.8 foi elaborado. Esse padrão é aplicável quando da necessidade de especificar um requisito que descreva uma transação de negócio, bem como o armazenamento dos dados pertinentes.

Quadro 3.8 – Padrão Processar Transação

Nome	Processar Transação
Domínio	Sistemas de Informação
Propósito	Processar e armazenar informação de uma transação.
Problema	Estabelecer quais dados da transação devem ser armazenados.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de processar uma transação e armazenar sua informação.
Tipo	Funcional
Solução	Template: Permitir o processamento de <transação> de <entidade>, com o armazenamento dos seguintes atributos: <atributos>.
	Parâmetro: <transação> nome da transação que contém informações do domínio (1). <entidade> nome da entidade que descreve o tipo de transação (1). <atributos> nome dos atributos da entidade que devem ser armazenados (1..*).
	Exemplo de Parâmetros: <transação > Venda <entidade> produto <atributos> número, data, cliente, produto, preço unitário e preço total.
	Exemplo da Descrição do Requisito: Permitir o processamento de venda de produto, com o armazenamento dos seguintes atributos: número, data, cliente, produto, preço unitário e preço total.
Padrões Relacionados	Informação Requerida Condição de Execução Limite de Execução Cálculo de Valor Recuperar Informação

3.3.3 Padrão de Requisito: Informações Gerenciais para SI

Nas organizações, os administradores e gerentes precisam de sistemas que auxiliem a monitoração, o controle, a tomada de decisão e as atividades administrativas. Nesse sentido, os Sistemas de Informações Gerenciais (SIG) proporcionam relatórios sobre o desempenho da organização (LAUDON; LAUDON, 2007). O objetivo principal do SIG é ajudar uma organização a alcançar seus objetivos, proporcionando por meio de relatórios, informações gerenciais detalhadas sobre as operações diárias da organização (STAIR; REYNOLDS, 2006).

O padrão Especificar Consulta Gerencial objetiva a escrita de requisitos que descrevem a necessidade de alguma informação gerencial sobre o negócio. As informações gerenciais requeridas pelas empresas, geralmente são referentes às transações processadas ou então sobre informações de cadastros básicos do negócio, como uma listagem de produtos, clientes, vendedores etc. Em ambos os casos, os dados estão armazenados em banco de dados. No Quadro 3.9 é mostrado o padrão Especificar Consulta Gerencial que atende a esses objetivos.

Quadro 3.9 – Padrão Especificar Consulta Gerencial

Nome	Especificar Consulta Gerencial
Domínio	Sistemas de Informação
Propósito	Relatório impresso ou em tela com informações de uma ou mais entidades.
Problema	Relacionar corretamente os atributos de cada entidade que devem constar do relatório a partir de uma determinada solicitação.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de realizar consultas de informações gerenciais em tela ou impresso.
Tipo	Funcional
Solução	<p>Template: Permitir a <operação> de <tipo> de <entidade> <condição>, contendo os seguintes atributos: <atributos>.</p> <p>Parâmetro: <operação> descrição da operação (1). <tipo> descrição do tipo de operação (1). <entidade> nome da entidade que contém informações do domínio (1). <condição> descrição da condição de execução da consulta (0..*).* <atributos> nome dos atributos da entidade que devem constar na consulta (1..*).*</p> <p>Exemplo de Parâmetros: <operação> impressão, listagem, consulta <tipo> um relatório, um comprovante, um histórico <entidade> venda <condição> com serviços em andamento, da data atual, de um período estabelecido, em atraso. <atributos> data, cliente e valor.</p> <p>Exemplo da Descrição do Requisito Permitir a impressão de um relatório de venda em um período estabelecido, contendo os seguintes atributos: data, cliente e valor. Permitir a impressão de um relatório de cliente, contendo os seguintes atributos: código, nome e endereço (sem o parâmetro condição).</p>
Padrões Relacionados	-

3.3.4 Padrões de Requisito: Regras de Negócio para SI

Os requisitos funcionais geralmente requerem a definição de regras de negócio específicas. Essas regras estabelecem restrições às operações do sistema. Como exemplo, uma regra de negócio pode estabelecer a política de desconto em vendas, as condições comerciais para vendas a prazo, o estabelecimento de um valor mínimo de um pedido, entre outras.

Para atender ao domínio de SI, foram elaborados quatro padrões de regras de negócio. O padrão Condição de Execução, apresentado no Quadro 3.10, tem por objetivo escrever uma regra de negócio que estabelece condições para a execução de uma operação específica. O padrão Cálculo de Valores ilustrado no Quadro 3.11, tem por finalidade a escrita de uma regra de negócio que descreve a necessidade da realização de um cálculo matemático, podendo ou não sua fórmula ser detalhada, bem como a respectiva condição para a execução do cálculo. O padrão Informação Requerida, mostrado no Quadro 3.12, objetiva a escrita de uma regra de negócio que estabelece as informações requeridas em uma determinada operação. O padrão Limite de Execução apresentado no Quadro 3.13, proporciona a escrita de uma regra de negócio que estabelece os limites de execução de uma operação do sistema.

Quadro 3.10 – Padrão Condição de Execução

Nome	Condição de Execução
Domínio	Sistemas de Informação
Propósito	Estabelecer as condições permitidas para uma determinada operação.
Problema	Especificar as condições específicas da operação.
Consequência	A escrita de um requisito que descreve uma regra de negócio que estabelece as condições de execução de uma determinada operação.
Tipo	Regra de Negócio
Solução	Template: A <operação> somente deve ser permitida se <condição>
	Parâmetro: <operação> nome da operação que deseja estabelecer uma condição(1). <condição > descrição da condição para a execução da operação (1..*).
	Exemplo de Parâmetros: <operação> reserva <condição> se houver estoque suficiente.
	Exemplos da Descrição do Requisito: A inclusão de um item na venda somente deve ser permitida se houver estoque suficiente. A reserva somente deve ser permitida se houver vagas suficientes para atendê-la. A remoção da reserva somente deve ser permitida se ocorrer antes do período prevista para a efetivação da venda. A venda somente deve ser permitida se o usuário for do nível de vendedor.
Padrões Relacionados	-

Quadro 3.11 – Padrão Cálculo de Valor

Nome	Cálculo de Valor
Domínio	Sistemas de Informação
Propósito	Especificar cálculos que transformam dados existentes em novos dados usando algoritmos ou fórmulas matemáticas específicas.
Problema	Estabelecer o cálculo matemática requerido em uma determinada operação do sistema.
Consequência	A escrita de um requisito que descreve a necessidade de um usuário de especificar um determinado cálculo matemático, requerido em uma operação.
Tipo	Funcional
Solução	Template: Permitir o cálculo <operação> <condição>.
	Parâmetro: <operação> descrição da operação que deseja calcular (1). <condição> descrição da condição para a execução do cálculo (1..*).
	Exemplo de Parâmetros: <operação> de horas extras dos funcionários <condição> que trabalharam mais que 8h. diárias
	Exemplo da Descrição do Requisito: Permitir o cálculo da data prevista para devolução com base na categoria do leitor. Permitir o cálculo de horas extras dos funcionários que trabalharam mais que 8h. diárias.
Padrões Relacionados	-

Quadro 3.12 – Padrão Informação Requerida

Nome	Informação Requerida
Domínio	Sistemas de Informação
Propósito	Determinar as informações requeridas em uma operação do sistema.
Problema	Estabelecer as entidades de dados necessárias para uma operação em questão.
Consequência	A escrita de um requisito que descreve uma regra de negócio que estabelece as informações requeridas por uma determinada operação.
Tipo	Regra de Negócio
Solução	Template: As informações de <entidade> deve(em) estar previamente cadastradas.
	Parâmetro: <entidade> nome da entidade que contém informações do domínio (1..*)
	Exemplo de Parâmetros: <entidade> cliente, produto, vendedor.
	Exemplo da Descrição do Requisito: As informações de cliente, produto e vendedor deve(em) estar(em) previamente cadastrada(s).
Padrões Relacionados	-

Quadro 3.13 – Padrão Limite de Execução

Nome	Limite de Execução
Domínio	Sistemas de Informação
Propósito	Determinar limites para uma determinada operação.
Problema	Estabelecer os limites específicos de uma operação.
Consequência	A escrita de um requisito que descreve uma regra de negócio que estabelece limites específicos de execução de uma determinada operação.
Tipo	Regra de Negócio
Solução	Template: A <operação> pode possuir <limite>
	Parâmetro: <operação> nome da operação que deseja estabelecer um limite (1). <limite> descrição do limite para a execução da operação (1..*).
	Exemplo de Parâmetros: <transação> venda <limite> muitos itens de produto, no máximo 2 itens por aluno.
	Exemplo da Descrição do Requisito: A venda de produto pode possuir muitos itens de produto. A reserva de obra pode possuir no máximo 2 itens por aluno.
Padrões Relacionados	-

Um apoio computacional foi desenvolvido para facilitar a elaboração de documentos de requisitos no domínio de SI. Com a utilização desse ferramental, o engenheiro de software passa a ter um guia que pode contribuir para o aumento da produtividade, a completude do DR. Na Figura 3.2 é apresentado um resumo dos padrões e seus relacionamentos.

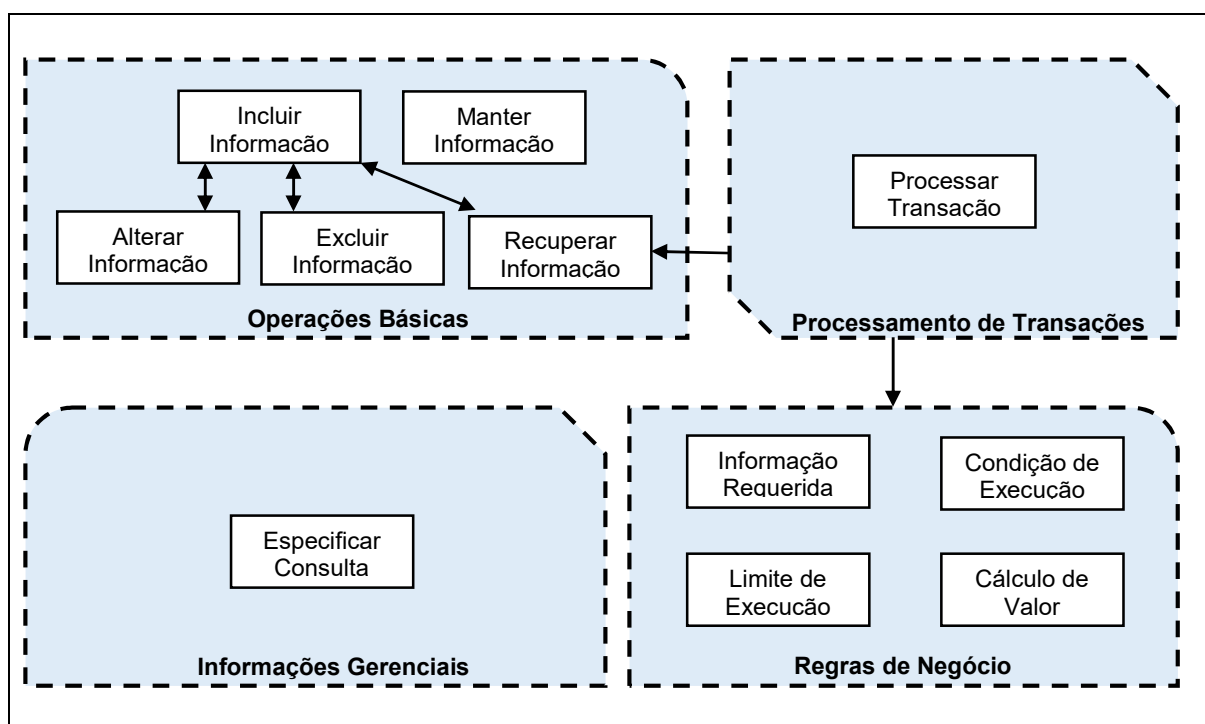


Figura 3.2 – Padrões de Requisitos para o Domínio de Sistemas de Informação

3.4 Considerações Finais

Os padrões apresentados orientam os engenheiros de software na identificação e escrita de requisitos funcionais no domínio de SI para a elaboração de um DR. Profissionais com experiência em engenharia de requisitos ou que possuem DR de projetos finalizados podem utilizar a metodologia definida neste trabalho para a elaboração de novos padrões de requisitos.

A elaboração dos padrões foi realizada a partir de DRs fictícios utilizados na academia como material didático de disciplinas e outros que estão disponibilizados na literatura especializada e na web. Cada requisito foi organizado pelo tipo de operação e, posteriormente, analisados para identificar se estavam completos e sem ambiguidade.

O engenheiro de software que utiliza esses padrões deve estar atento a todas as sugestões de padrões relacionados existentes, o que pode ser difícil de ser realizado sem um apoio computacional. A partir dessa premissa, um apoio computacional foi desenvolvido para auxiliar o engenheiro de software na identificação e escrita de requisitos, que é apresentado no próximo capítulo, bem como o processo de utilização dos padrões.

Capítulo 4

APOIO COMPUTACIONAL PARA A ELABORAÇÃO DE DOCUMENTO DE REQUISITOS COM PADRÕES

4.1 Considerações Iniciais

Palomares et al. (2014) afirmam que há dificuldade de utilizar padrões sem um apoio computacional para direcionar o engenheiro de software. Esses autores apresentaram o resultado preliminar da pesquisa cujo objetivo foi sobre o uso atual das práticas de reutilização na engenharia de requisitos. Em especial, também relatam a opinião dos participantes sobre o reuso com base em padrões de requisitos de software. Um dos quesitos avaliados foi quanto aos fatores críticos para a introdução de um catálogo de padrões de requisitos. Dois pontos principais foram citados: a necessidade de um método de uso bem definido e uma ferramenta de suporte.

Neste trabalho, após criar os padrões de requisitos, considerando as observações de Palomares et al. (2014) e considerando que a usabilidade dos padrões fica facilitada, foi também desenvolvido o processo de uso e um apoio computacional a fim de auxiliar engenheiros de software na atividade de elicitação de requisitos com a utilização dos padrões elaborados.

O processo de uso consiste na utilização do apoio computacional para a instanciação dos padrões a fim de elaborar DRs de SI completos e consistentes.

Na Seção 4.2 é abordada a construção do apoio computacional. Na Seção 4.3 é apresentado o processo de uso do apoio computacional com os padrões de requisitos para a elaboração do DR. Na Seção 4.4 são descritas as funções do apoio

computacional. Na Seção 4.5 é apresentado um exemplo de aplicação do apoio computacional. As considerações finais são apresentadas na Seção 4.6.

4.2 Construção do Apoio Computacional

Para o desenvolvimento do apoio computacional, foi seguido o modelo de processo iterativo e incremental, o que permitiu melhorar as funcionalidades, a usabilidade e a inclusão de novas ideias, durante o desenvolvimento do projeto.

A linguagem orientada a objetos Java foi escolhida, envolvendo o desenvolvimento do código fonte na plataforma Java SE (*Standard Edition*), com o banco de dados MySQL *Server Community*, a biblioteca JDBC (*Java Database Connectivity*) do MySQL (*mysql-connector-java-5.1.23-bin.jar*) para permitir a conexão com o banco de dados e a biblioteca iText (*itext-2.1.7.jar*) para a criação do DR no formato *PDF*. Na Figura 4.1 é apresentado o modelo conceitual do apoio computacional.

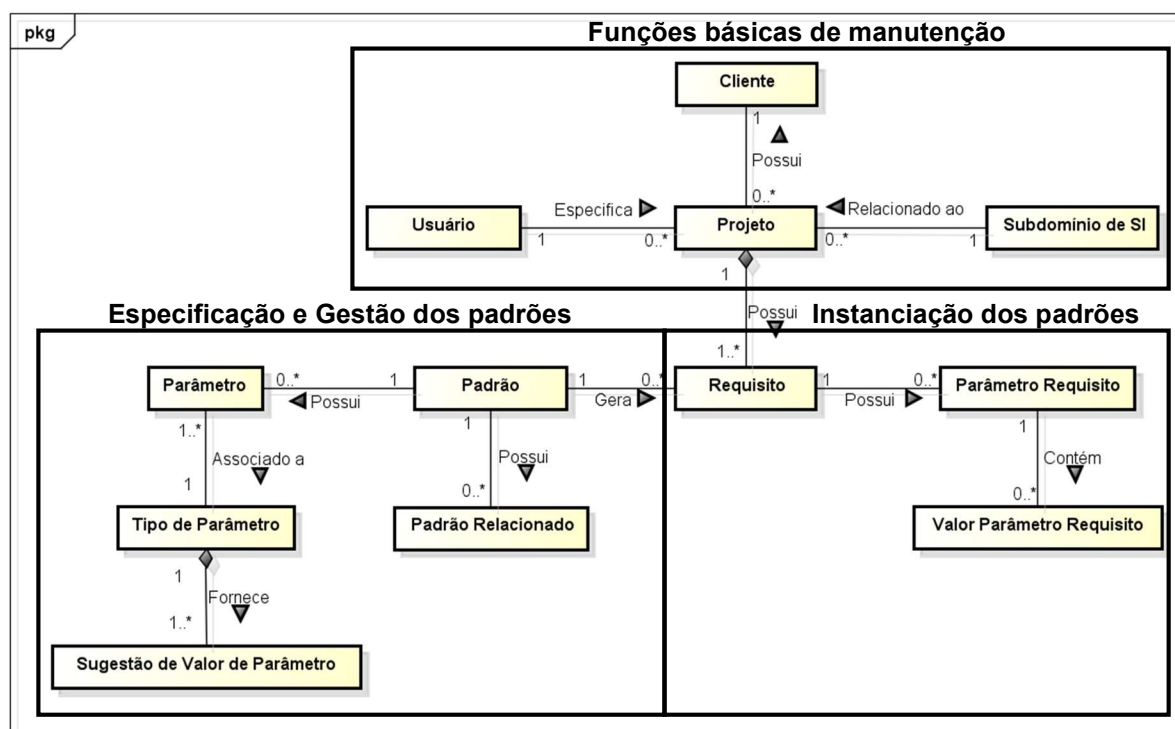


Figura 4.1 – Modelo Conceitual do Apoio Computacional

Um padrão (Padrão) pode possuir padrões relacionados (Padrão Relacionado), que contribuem para a completa elaboração do DR. Cada padrão pode possuir ou não parâmetros (Parâmetro) correspondentes à parte variável da solução proposta pelo padrão. Caso exista, deve estar associado a um tipo (Tipo de Parâmetro) que por sua vez fornece sugestões de valores de preenchimento para os parâmetros (Sugestão de Valor de Parâmetro) para ajudar o engenheiro de software durante a escrita da parte variável do requisito.

Para a escrita de um requisito (Requisito), deve existir um projeto (Projeto), que requer a seleção do cliente (Cliente), do usuário (Usuário) responsável pela especificação do DR e do subdomínio de SI (Subdomínio de SI) que orienta o engenheiro de software no reuso dos requisitos especificados em projetos futuros. O *template* da solução do padrão (Padrão) é fornecido para a especificação do requisito (Requisito), que possui uma parte do texto fixa e outra variável. A parte variável é chamada de parâmetro do requisito (Parâmetro Requisito) que foi fornecida pelo padrão para receber os valores informados (Valor Parâmetro Requisito) pelo engenheiro de software.

Os requisitos gerados em projetos concluídos ficam armazenados em um repositório e podem ser reusados para a escrita de requisitos em novos projetos, com a possibilidade de alteração, se necessário.

4.3 Processo de Uso do Apoio Computacional

O engenheiro de software ao utilizar o apoio computacional para a escrita de requisitos, a fim de elaborar o DR, deve executar os seguintes passos:

1. **Seleção do Padrão:** de acordo com a necessidade do cliente, o padrão adequado é escolhido e essa escolha sempre é confirmada com o cliente. Se o padrão atende aos requisitos solicitados, o engenheiro de software passa para o passo seguinte.
2. **Especificação das partes variáveis:** o engenheiro de software informa ao cliente as partes variáveis da solução do padrão selecionado. O cliente pode fornecer as informações pertinentes a partir do conjunto de sugestões cadastradas para o padrão no apoio computacional. No entanto, existe

também a possibilidade de o engenheiro de software realizar esse passo, sugerindo ao cliente possíveis soluções de requisitos que foram anteriormente utilizadas em outros projetos e que estão disponíveis no repositório.

3. **Especificações complementares:** o engenheiro de software identifica junto ao cliente a existência de alguma regra de negócio da operação em questão e/ou algum requisito funcional complementar. Em ambos os casos um padrão pode ser utilizado para complementar a especificação. Dessa forma, volta-se aos passos 1 e 2 para a especificação dos requisitos complementares. O engenheiro de software também pode propor algum requisito complementar por meio das sugestões de padrões relacionados fornecidas pelo apoio computacional.

4.4 Funções Implementadas no Apoio Computacional

O apoio computacional para a elaboração do DR no domínio de SI possui três módulos: i) o de especificação e gestão dos padrões, que geralmente fica sob a responsabilidade de um engenheiro de software com mais experiência; ii) o de instanciação dos padrões durante a elicitação de requisitos; e iii) o de funcionalidade básica, inerente à manutenção de usuário, cliente, projeto e sugestão de parâmetro.

Na Figura 4.2 é apresentada a interface de usuário, com as funções básicas, que têm por objetivo a manutenção de usuários que podem ter acesso ao apoio computacional. Os usuários com o perfil de administrador têm acesso a todas as funcionalidades, já os demais têm acesso apenas à instanciação de padrões. Para ter acesso ao sistema, os usuários devem ter um login (*Nome de Usuário*), *Senha* e o *Status* “Ativado”.

The 'Usuário' window contains the following fields: Código, Nome, Sobrenome, CPF, Data de Nascimento (with a date picker), Cargo, Endereço, Cidade, Estado, Telefone, Perfil Administrador (dropdown menu), Nome de Usuário, Senha, and Status (dropdown menu). Below the form are buttons for 'Novo', 'Salvar', 'Cancelar', 'Alterar', and 'Excluir'. At the bottom is a table with the following data:

Código	Nome	Sobrenome	Cargo	Telefone	Nome de Usuário	Status
4	Leonardo	Barcelos	Engenheiro de Software	(34) 3423-2700	leonardo	Ativado
9	Rosangela	Penteado	Analista de Requisitos	(16) 3351-8621	rosangela	Ativado

Figura 4.2 – Módulo de Manutenção de Usuário

Na Figura 4.3 é ilustrada a interface de cliente, que é usada para a manutenção de clientes, pessoa física ou jurídica, que contrataram o desenvolvimento de um projeto de software.

The 'Cliente' window includes a 'Tipo' section with radio buttons for 'Pessoa Jurídica' (selected) and 'Pessoa Física'. The form fields are: Código, Nome, Endereço, Complemento, Bairro, Cidade, Estado, CEP, CNPJ, Contato, Telefone 1, Telefone 2, Celular, e-mail, and Observação. Below the form are buttons for 'Novo', 'Salvar', 'Cancelar', 'Alterar', and 'Excluir'. At the bottom is a table with the following data:

Código	Nome	Telefone	email
10	Universidade Federal de São Carlos	(16) 3351-8233	secppgcc@dc.ufscar.br

Figura 4.3 – Módulo de Manutenção de Cliente

A interface de manutenção de projetos (Figura 4.4), objetiva registrar projetos de software em desenvolvimento, para posteriormente permitir a escrita dos requisitos com o uso dos padrões. Os campos que devem ser preenchidos são: *Nome* que é usado para fornecer um nome ao projeto; *Visão Geral*, que fornece uma visão geral da funcionalidade do projeto. A informação constante nesse campo fará parte do DR.

Para o registro do projeto, é necessário que o *Cliente* e o usuário *Responsável* pela escrita dos requisitos estejam cadastrados no sistema. O campo *Status* define se o projeto está em aberto ou finalizado e o campo *Subdomínio*, determina o tipo do sistema em desenvolvimento no contexto de SI. Exemplo: Sistema de Hotel, Sistema de Biblioteca, Sistema de Locação, etc. O subdomínio contribui para a orientação do engenheiro de software durante a reutilização de requisitos de projetos anteriores. Assim, se um Sistema de Hotel for desenvolvido, o engenheiro de software pode reaproveitar alguns dos requisitos já elicitados para sistemas anteriores do mesmo tipo.

A interface do usuário para a manutenção de projetos, intitulada 'Projeto'. Ela contém os seguintes campos de formulário:

- Código: campo de texto.
- Nome: campo de texto.
- Visão Geral: campo de texto com uma barra de rolagem.
- Cliente: menu suspenso.
- Responsável: menu suspenso.
- Data: campo de texto com máscara ' / '.
- Status: menu suspenso.
- Sub Domínio: menu suspenso.

Abaixo dos campos, há uma barra de ações com os botões: Novo (com ícone de plus verde), Salvar (com ícone de disquete), Cancelar (com ícone de X vermelho), Alterar (com ícone de lápis) e Excluir (com ícone de lixeira).

Na base da interface, há uma tabela com as seguintes colunas: Código, Nome, Cliente, Responsável, Data e Status.

Código	Nome	Cliente	Responsável	Data	Status
24	Sistema Hotel Uirapuru	Universidade Federal de São Carlos	Leonardo	05/01/2015	Aberto

Figura 4.4 – Módulo de Manutenção de Projeto

Na Figura 4.5 é mostrada a interface para a inclusão de sugestões para a parte variável no padrão. Conforme mencionado anteriormente, a parte variável é denominada de parâmetro e deve estar associada a algum tipo. Essa associação proporciona o fornecimento de um conjunto de sugestões para o preenchimento do parâmetro. No exemplo da Figura 4.5, a interface mostra um tipo denominado *entidadeAtributos*, que tem por objetivo fornecer um conjunto de possíveis atributos encontrados em entidades de dados. Esse tipo possui sugestões de atributos para

diferentes entidades de dados, como, Acomodação, Carro, Categoria, Devolução etc. No exemplo, a entidade *Pessoa* está apresentando os possíveis atributos que podem ser usados, por exemplo, para a escrita de um requisito que descreve um cadastro de cliente. As sugestões disponíveis são: *codigo*, *nome*, *tipo*, *data de nascimento*, *endereço*, *cidade*, *estado*, *país*, *passaporte*, *telefone fixo*, *celular*, *e-mail*, *cnpj*, *cpf*, *RG* e *nível de acesso*.

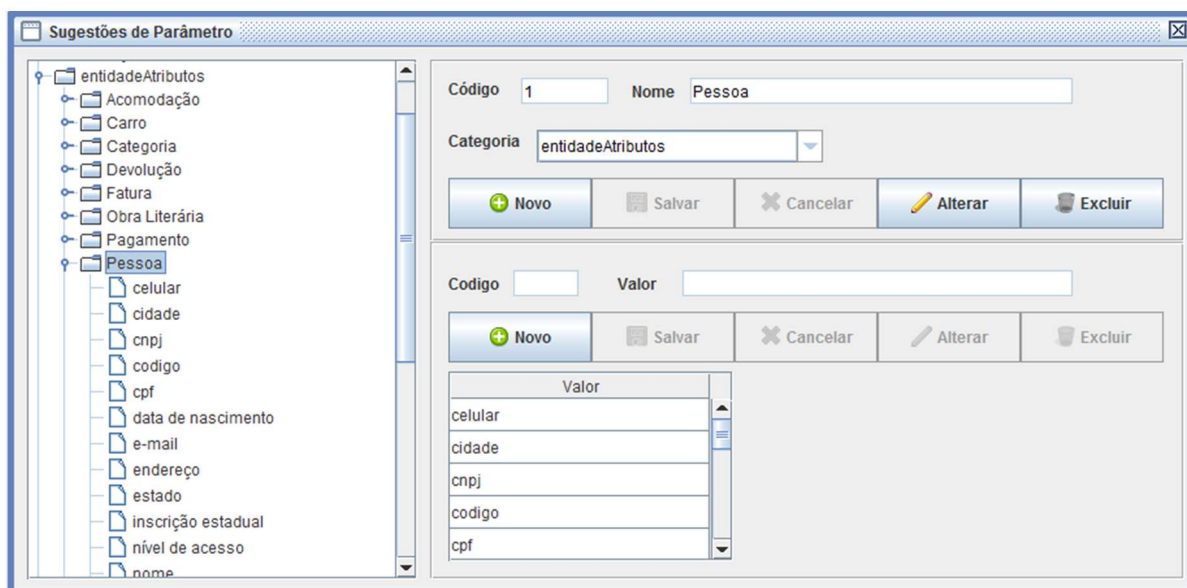


Figura 4.5 – Módulo de Manutenção de Sugestões de Parâmetros

Com o módulo de especificação e gestão de padrões (Figura 4.6) é possível realizar a manutenção dos padrões. A aba Informações Básicas é utilizada para a especificação do padrão, sendo necessário o preenchimento dos elementos: *nome*, *domínio*, *propósito*, *problema*, *consequência*, *tipo* e *solução*. Após o preenchimento, o botão [*Extrair Parâmetros*] deve ser selecionado para estabelecer para cada parâmetro da solução: a) um *tipo*, que está associado a um conjunto de sugestões de preenchimento que pode ser usado na instanciação do padrão; b) uma *descrição* que orienta o correto preenchimento do parâmetro; c) a *multiplicidade* que orienta a quantidade de valores permitidos para o parâmetro.

Os padrões são apresentados em uma estrutura de árvore organizada pelo tipo de padrão, que, atualmente, podem representar requisitos funcionais e regras de negócio.

Para a especificação dos padrões relacionados ao padrão em elaboração, a aba Padrões Relacionados deve ser utilizada. Um padrão pode ser relacionado com quaisquer outros padrões criados.

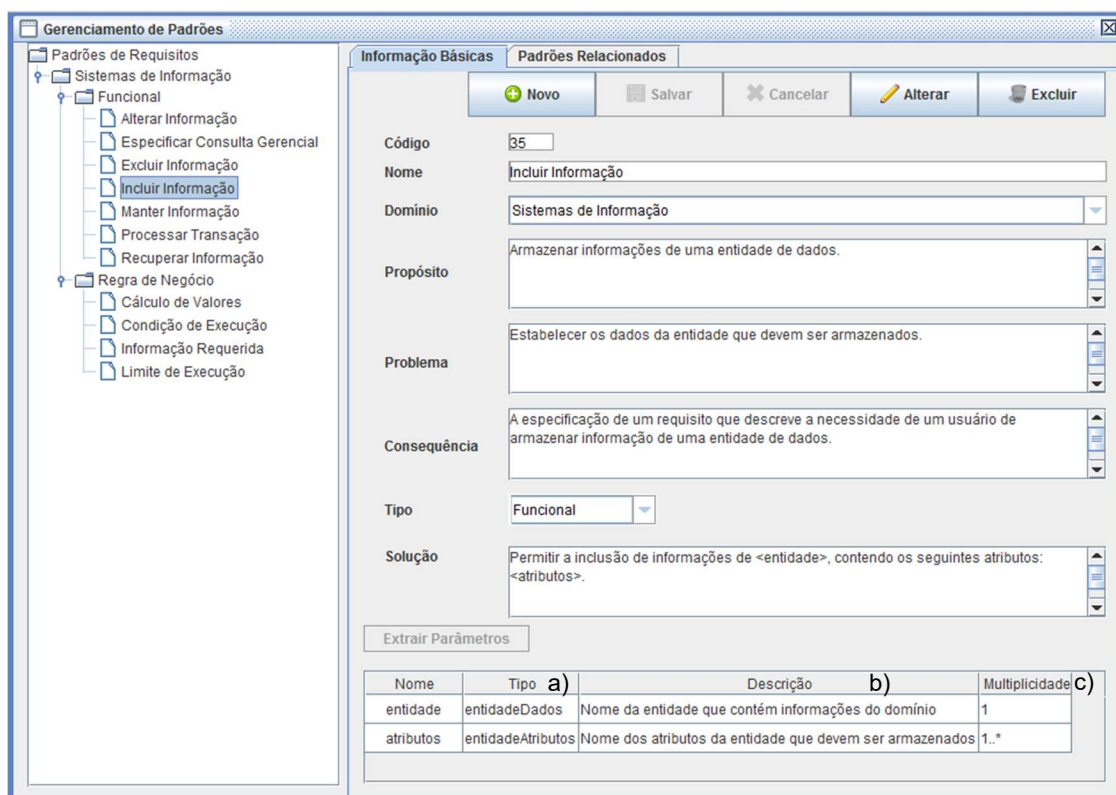


Figura 4.6 – Módulo de Especificação e Gestão de Padrões

O módulo que realiza a instanciação de padrões possui quatro abas, que são apresentadas em detalhes nos próximos tópicos:

- **Aba Informações Básicas:** apresenta as informações de aplicabilidade e solução do padrão (Figura 4.7). Caso o padrão atenda às necessidades, o botão [*Instanciar*] deve ser pressionado. No exemplo da Figura 4.7, o Padrão Incluir Informação está sendo apresentado e pode ser instanciado para a escrita de um requisito para o armazenamento de informação de uma entidade de dados.

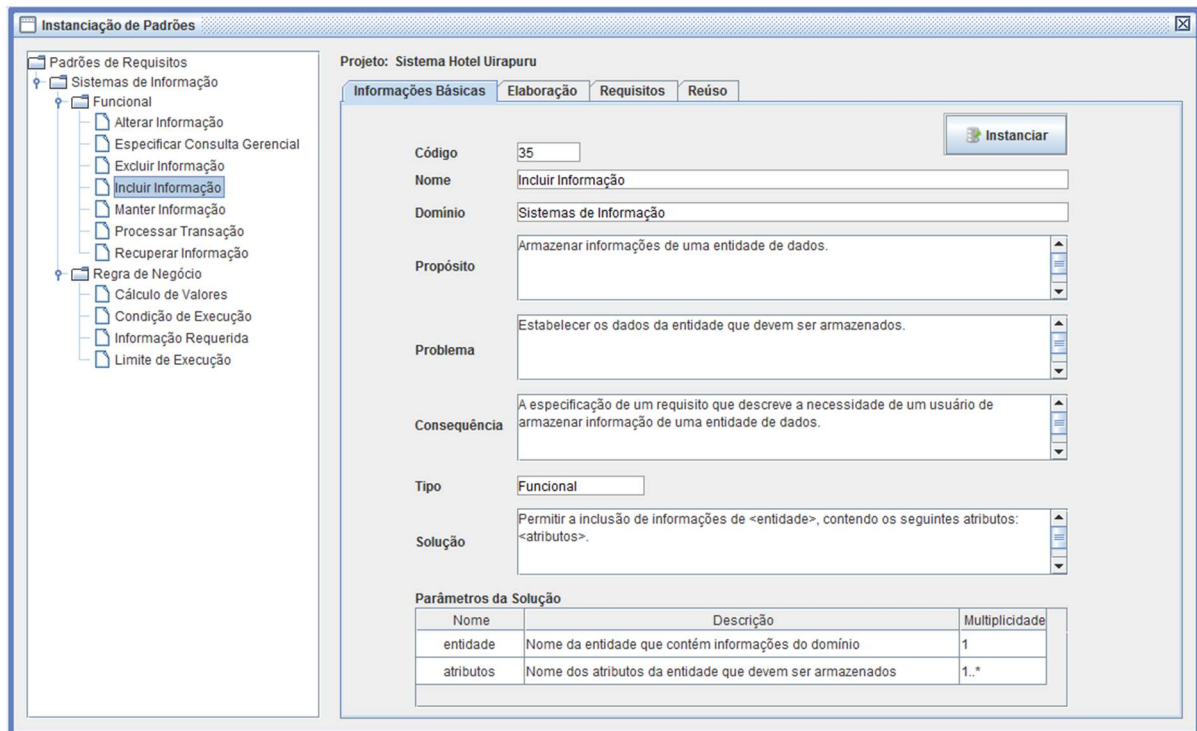
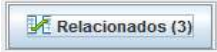


Figura 4.7 – Módulo de Instanciamento de Padrões – Aba Informações Básicas

- **Aba Elaboração:** usada para a escrita de um requisito por meio do padrão escolhido (Figura 4.8), sendo que as letras colocadas nessa figura são explicadas a seguir: a) um campo para descrição do requisito; b) o *template* da solução; c) uma lista com os parâmetros; d) uma lista de sugestões organizada em categorias de sugestões; e) ao selecionar a categoria de sugestão, uma lista de possíveis valores é apresentada.

O Padrão Incluir Informação está sendo instanciado para a inclusão de clientes, como pode ser visto na Figura 4.8. Nesse exemplo, o parâmetro <atributos> é preenchido com valores que correspondem aos atributos de cliente. Com a seleção desse parâmetro são fornecidas algumas categorias de sugestões, como *venda*, *produto*, *pessoa* etc. Ao selecionar uma dessas sugestões, outras são apresentadas para o preenchimento do parâmetro. Após essas seleções, o botão [+] deve ser pressionado para que os valores selecionados sejam atribuídos ao parâmetro. Caso seja necessário um valor de preenchimento diferente dos sugeridos, a especificação desses novos valores é feita por meio do botão [Novo]. Após a finalização da escrita de um requisito, ou seja, com o preenchimento de todos os parâmetros, o botão [Salvar] deve ser pressionado.

Ao finalizar a escrita do requisito, como o padrão Incluir Informação possui padrões relacionados, o botão  mostra um número que corresponde à quantidade de padrões que podem complementar a escrita do requisito em questão. Ao clicar, são fornecidos os padrões relacionados com suas respectivas informações de aplicabilidade e solução, podendo ou não serem instanciados.

Existe também a possibilidade de escrever um requisito sem a utilização de um padrão. Para isso é necessário usar o botão [Novo] que deixará o campo *Template* em branco para a escrita do novo requisito.

Na instanciação de um padrão relacionado é definida automaticamente a relação de dependência do requisito em questão com o requisito que foi instanciado pelo padrão relacionado.

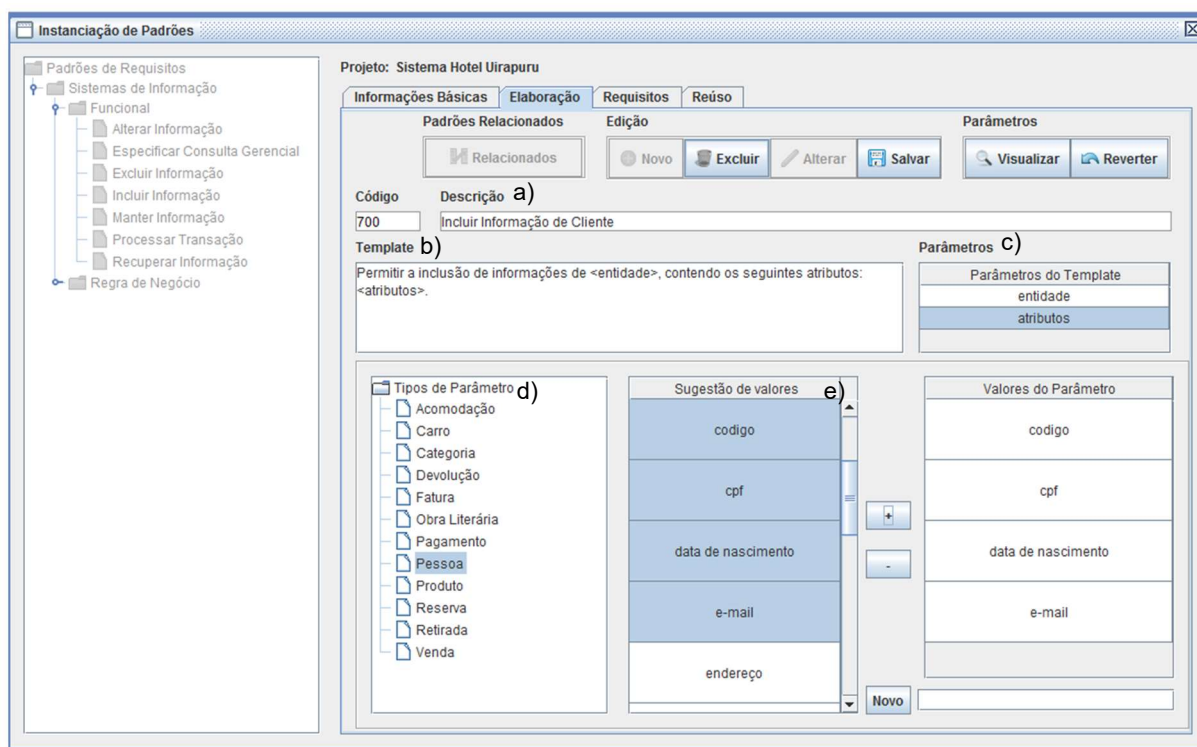


Figura 4.8 – Módulo de Instanciação de Padrões – Aba Elaboração

- **Aba Requisitos:** apresenta os seguintes itens: a) todos os requisitos do projeto. Ao selecionar um requisito também são apresentados os requisitos relacionados, caso existam requisitos relacionados; b) permite a seleção do requisito para a edição, visualização ou exclusão. A edição ocorre na aba elaboração; c) fornece sugestões de padrões relacionados com o requisito selecionado; d) permite o estabelecimento de relacionamento

(rastreamento) entre os requisitos; e) permite a geração do documento de requisitos com todos os requisitos, no formato *PDF*. Na Figura 4.9 é apresentada a interface da aba Requisitos.

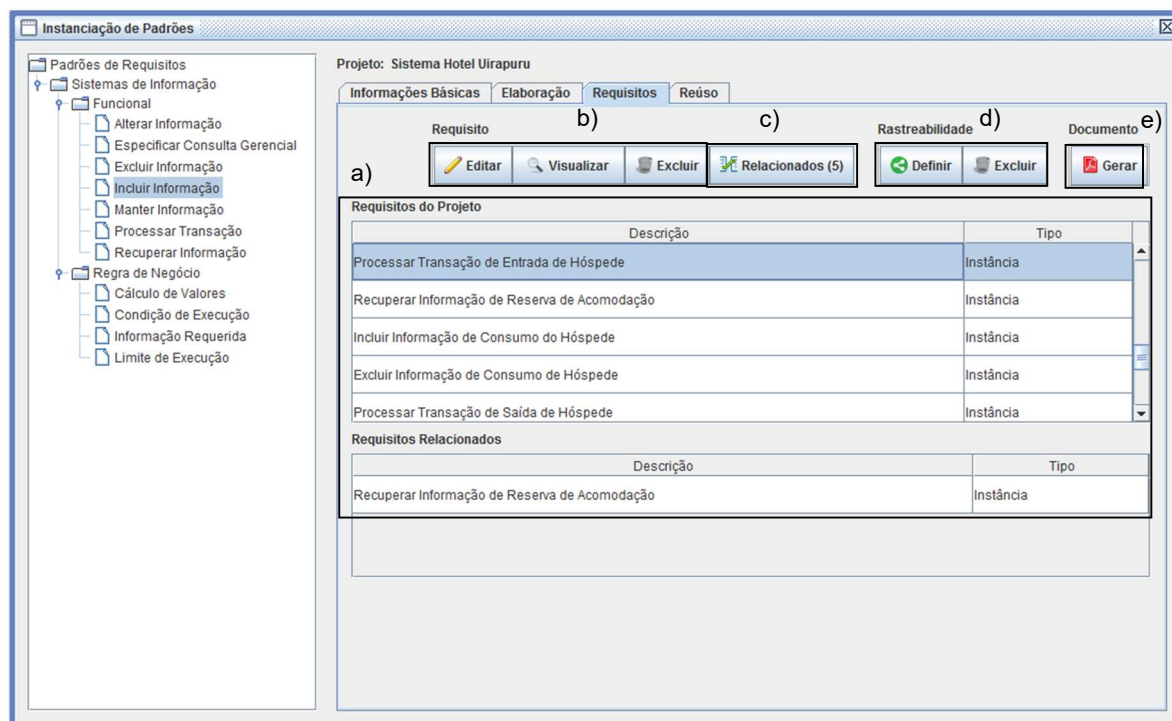


Figura 4.9 – Módulo de Instanciação de Padrões – Aba Requisitos

- **Aba Reúso:** apresenta os requisitos do repositório que foram instanciados em outros projetos, referentes ao padrão selecionado. No exemplo que é apresentado na Figura 4.10, como o padrão Incluir Informação está selecionado, estão sendo apresentadas sugestões de requisitos que foram instanciados com esse padrão em projetos de “reserva de bens e serviços” e “restaurante”. Esses requisitos podem ser reusados no projeto atual integralmente ou com a possibilidade de ter alterações. Existem duas possibilidades de reúso, sendo a de simplesmente reusar o requisito selecionando o botão [*Reusar*] ou reusar com os requisitos dependentes (relacionados) selecionando o botão [*Reusar com Dependentes*]. Para conferir a especificação do requisito selecionado, o botão [*Visualizar*] possui essa funcionalidade.

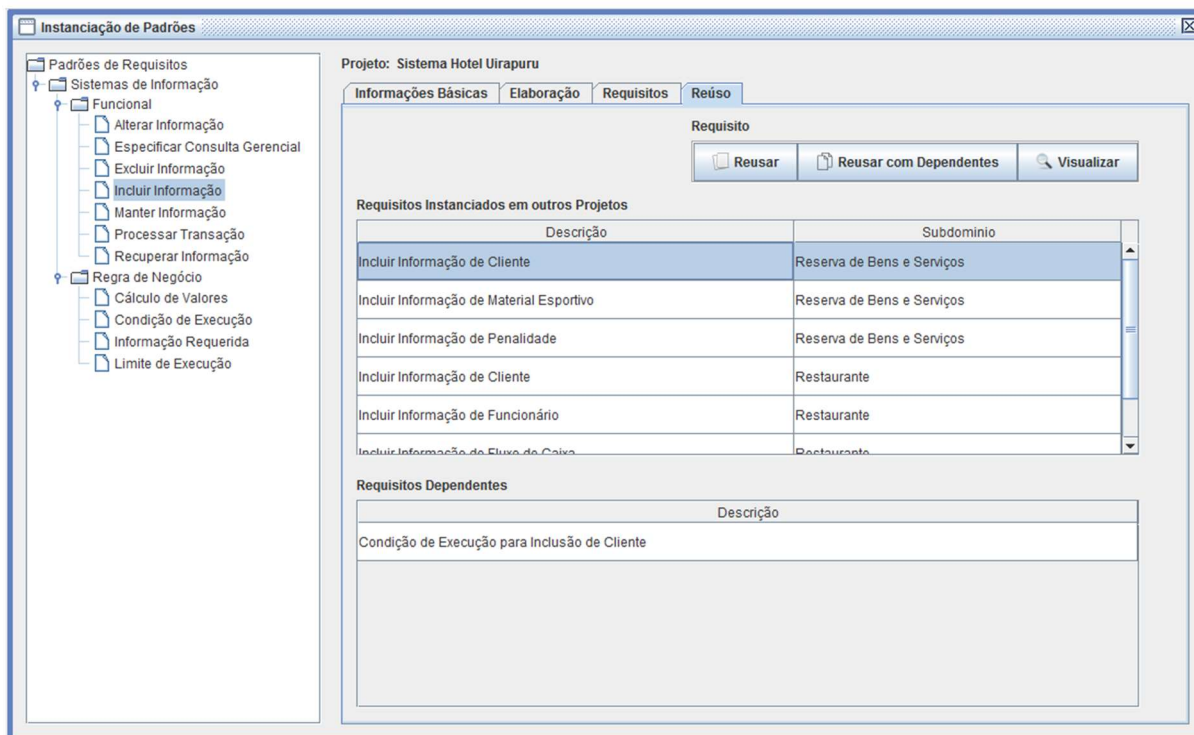


Figura 4.10 – Módulo de Instanciação de Padrões – Aba Reúso

Uma versão simplificada do documento de requisitos resultante da instanciação dos padrões apresentados anteriormente é apresentada na Figura 4.11. Os requisitos funcionais são identificados pela sigla “RF” e as regras de negócio pela sigla “RN”. A coluna da tabela “Depende de” apresenta a dependência entre os requisitos. No exemplo em questão, o requisito RF2 depende do requisito RF10. Um exemplo de aplicação do apoio computacional bem como uma versão completa do DR é apresentação na Seção 4.5.

Identificador:	Descrição	Depende de:
RF1	Permitir a inclusão de informações de Cliente, contendo os seguintes atributos: código, nome, endereço e cidade.	
RF2	Permitir o processamento de Venda de Produto, com o armazenamento dos seguintes atributos: código, data da venda, cliente, produto, quantidade, valor unitário e valor total. RN2.1. A venda de produto somente deve ser permitida se houver estoque suficiente	RF10

Figura 4.11 – Exemplo de Requisitos Especificados pelo Apoio Computacional

4.5 Exemplo de Aplicação do Apoio Computacional

A fim de exemplificar o uso do apoio computacional, bem como um DR elaborado com o uso dos padrões de requisitos, nesta seção é apresentado um exemplo de aplicação na especificação de um sistema de videolocadora, como apresentado no Quadro 4.1.

Quadro 4.1 – Visão Geral do Sistema de Videolocadora

Visão Geral	
O cliente vai até a videolocadora para alugar um filme. Ao chegar no balcão, informa o título do filme para consulta. O sistema deve informar a existência do filme e se está disponível. Se o cliente quiser alugar o filme, o empréstimo é registrado no sistema. O empréstimo é realizado somente se o filme estiver disponível. Se o filme não estiver disponível, o cliente poderá efetuar uma reserva. Para a efetivação do empréstimo do filme reservado, o sistema deve ser capaz de recuperar a reserva previamente realizada.	
Atributos Necessários nas Entidades de Dados do Sistema de Videolocadora	
<u>Filme</u>	código, título, categoria e status.
<u>Cliente</u>	código, nome, endereço, cidade, estado, telefone 1, telefone 2, celular, rg, cpf e e-mail.
<u>Empréstimo</u>	código, identificação do cliente, identificação do filme, data do empréstimo, data da devolução, valor.
<u>Reserva</u>	código, identificação do cliente, identificação do filme, data da reserva e data da retirada.

4.5.1 Elaboração do Documento de Requisitos

A elaboração do DR por meio do apoio computacional inicia-se com o cadastro do projeto, sendo necessário fornecer: um nome para o projeto, uma visão geral, a seleção do cliente e analista responsáveis (previamente cadastrados), a data de início do projeto, o status, que inicialmente deve ser deixado como “aberto” para indicar que o projeto está em andamento e a seleção do subdomínio de SI, para orientar o engenheiro de software na reutilização futura dos requisitos desse projeto em outros projetos. Na Figura 4.12 é apresentado o cadastro do projeto do sistema de videolocadora.

Código	Nome	Cliente	Responsável	Data	Status
24	Sistema Hotel Uirapuru	Universidade Federal de São Carlos	Leonardo	05/01/2015	Aberto
27	Sistema da Biblioteca C...	Universidade Federal de São Carlos	Leonardo	01/06/2015	Finalizado
28	Sistrema de Almoxafad...	Universidade Federal de São Carlos	Leonardo	01/06/2015	Finalizado
29	Sistema de Gerenciame...	Universidade Federal de São Carlos	Leonardo	01/06/2015	Finalizado
30	Sistema de Locação de ...	Universidade Federal de São Carlos	Leonardo	01/06/2015	Finalizado

Figura 4.12 – Cadastro do Projeto de Videolocadora

Para a escrita dos requisitos do projeto de Videolocadora, a função de *Instanciar* do menu *Padrão* deve ser selecionada (Figura 4.13). Em seguida, uma caixa de diálogo solicitando a escolha do projeto é apresentada, ou seja, são apresentados todos aqueles que estão com o status “Aberto”. Após a escolha, o botão [*Selecionar*] deve ser pressionado.

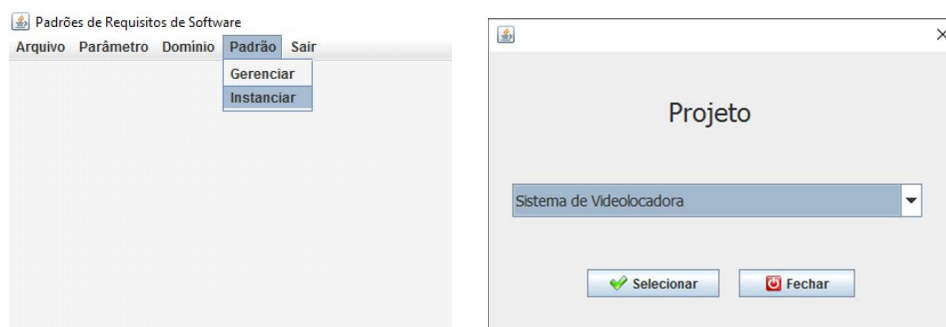


Figura 4.13 – Seleção do Projeto de VideoLocadora

O primeiro passo consiste na seleção do padrão de acordo com as necessidades declaradas pelos *stakeholders*. Analisando a declaração inicial na visão geral de exemplo (Quadro 4.1), entende-se que uma solução adequada para o

problema é que tanto filme quanto cliente devem existir no banco de dados, para facilitar as operações da videolocadora.

Nesse sentido, o padrão Incluir Informação pode ser reutilizado para a escrita dos requisitos de inclusão de filme e cliente. Na Figura 4.14 é apresentada a interface de instanciação do padrão Incluir Informação selecionado, mostrando a sua respectiva descrição. O botão [*Instanciar*] deve ser pressionado para iniciar a escrita do requisito com esse padrão. Ao pressionar o referido botão, a aba *Elaboração* é selecionada automaticamente para iniciar a escrita.

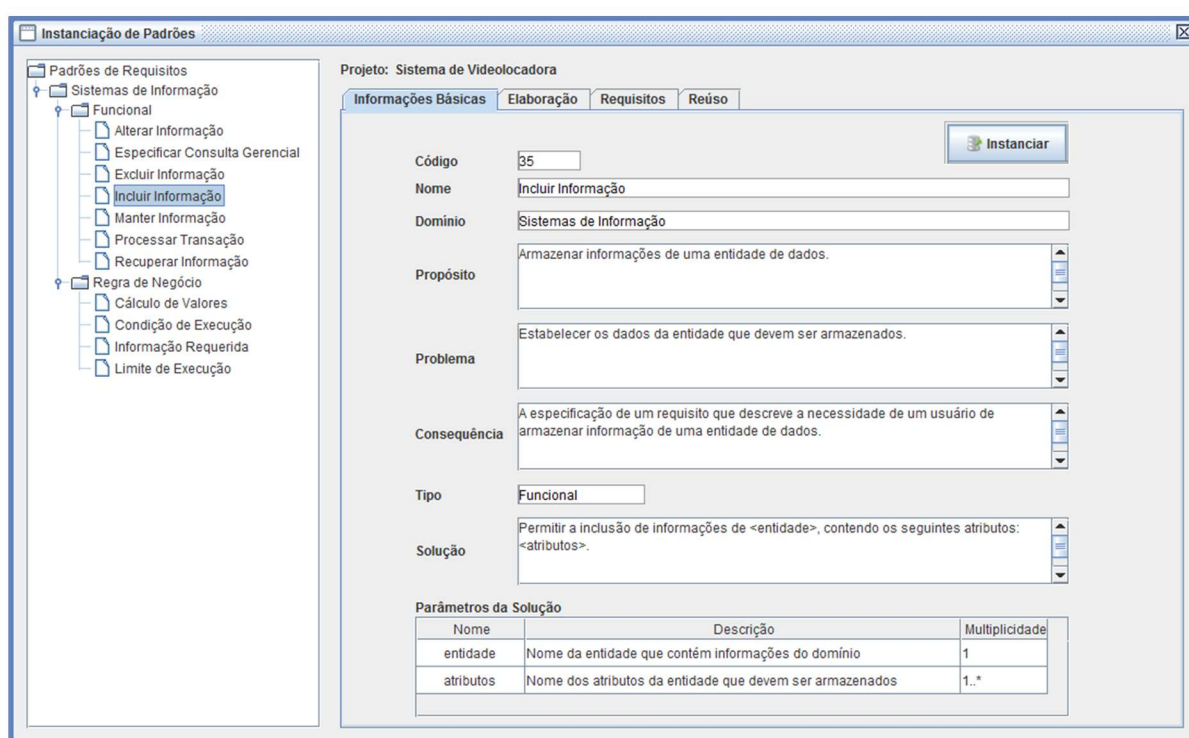


Figura 4.14 – Instanciamento do Padrão Incluir Informação

Na Figura 4.15 são apresentados os passos para a escrita do requisito de inclusão de filme, sendo que a numeração colocada nessa figura é explicada a seguir:

- a) Definição de uma *Descrição* para o requisito, para facilitar a sua localização. O próprio apoio computacional fornece inicialmente o nome do padrão como sugestão de descrição (incluir informação), bastando ser completada com a entidade de dados a ser tratada. Entretanto, pode ser definida qualquer descrição. Para o exemplo de inclusão de filme, a descrição fornecida foi *Incluir Informação de Filme* (a1);
- b) No *Template* (b1) é apresentado o texto da solução com os parâmetros *entidade* e *atributos*, que devem ser preenchidos de acordo com as

informações específicas descritas na visão geral de exemplo. Para atribuir valores aos parâmetros, eles devem ser selecionados na tabela *Parâmetros* (b2) localizada a direita do *template*;

- c) Com o parâmetro *entidade* selecionado para o seu preenchimento (c1). O engenheiro de software possui duas opções: a escrita manual do nome da entidade e clicar no botão [*Novo*] (c2) ou então a seleção do nome da entidade a partir das sugestões fornecidas pelo repositório do apoio computacional (c3). Para a seleção das sugestões, primeiro deve ser selecionado um dos tipos de parâmetros, que nesse caso, existe apenas um, sendo o *Entidade*, que objetiva o fornecimento de sugestões de entidades de dados. As sugestões são apresentadas na tabela de *Sugestões de valores* (c4). Nesse exemplo, a sugestão *Filme* foi escolhida e o botão [+] foi pressionado para atribuir “Filme” ao parâmetro *entidade*. O valor selecionado é inserido na tabela *Valores do Parâmetro* (c5);
- d) Com a seleção do botão [*Visualizar*] (d1) o parâmetro no *template* é substituído pelo valor atribuído ao parâmetro. Esse passo não é obrigatório, caso não seja realizado, a substituição ocorrerá quando o botão [*Salvar*] for pressionado para concluir a escrita do requisito;
- e) O passo c) deve ser novamente realizado para o preenchimento do parâmetro *atributos* (e1). O resultado desse passo por meio do uso das sugestões de valores é apresentado na Figura 4.16;
- f) Finalmente, o botão [*Salvar*] (f1) deve ser pressionado para concluir a escrita do requisito.

Com a execução desses passos, o requisito Incluir Informação de Filme é registrado no projeto da videolocadora.

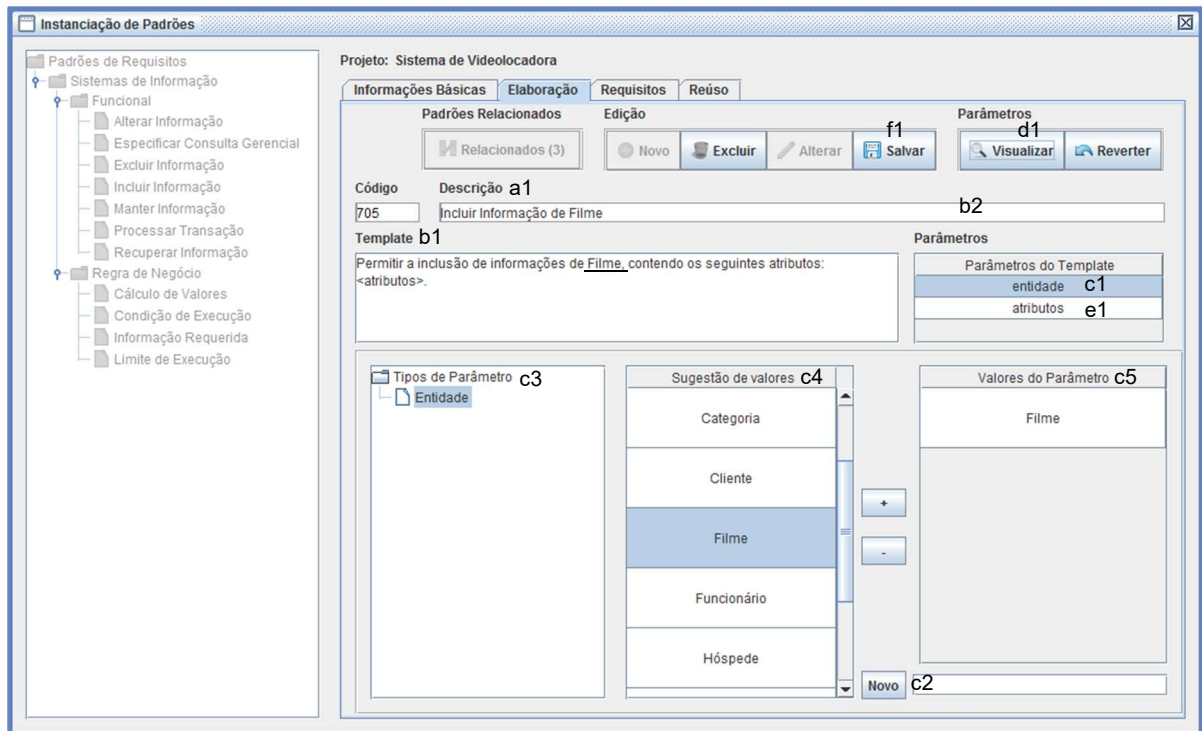


Figura 4.15 – Escrita do Requisito para Incluir Informação de Filme: Preenchimento do Parâmetro entidade

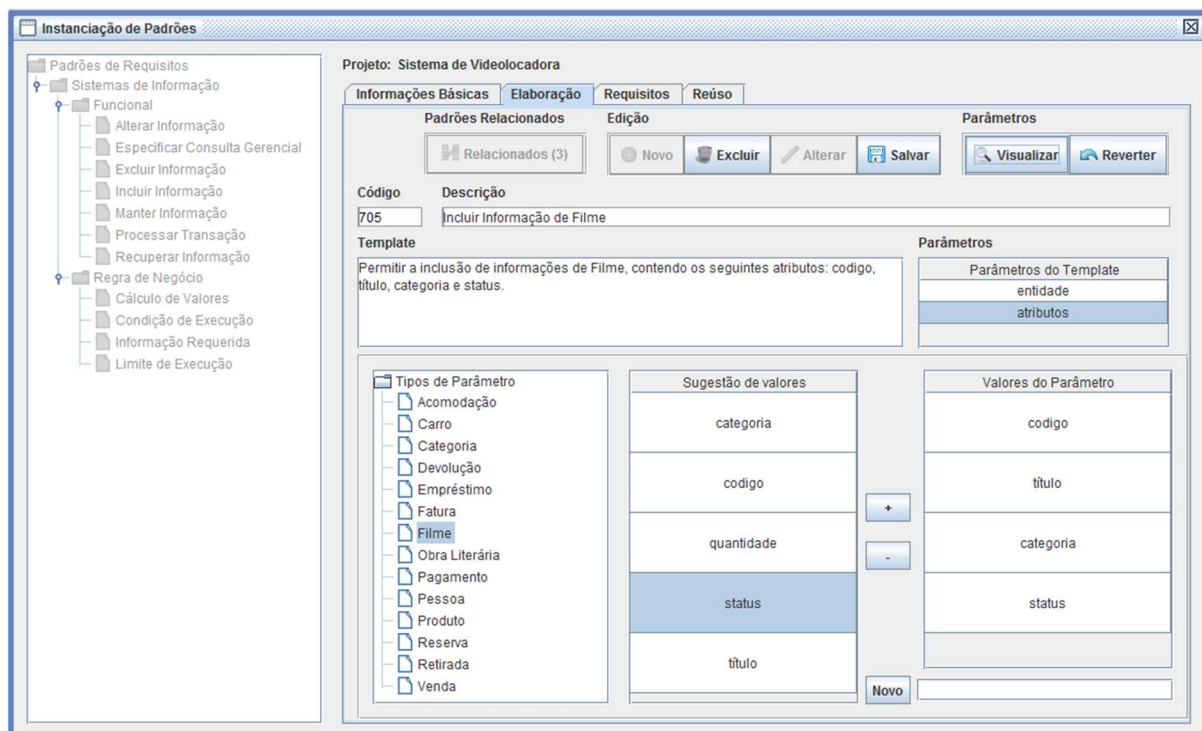


Figura 4.16 – Escrita do Requisito para Incluir Informação de Filme: Preenchimento do Parâmetro atributos

Ao salvar um requisito o apoio computacional pode sugerir ao engenheiro de software algum padrão relacionado, se existir, ao padrão instanciado para

complementar a especificação do requisito recém-escrito. Essa sugestão é realizada por meio do botão [Relacionados] que apresenta um número que corresponde a quantidade de padrões relacionados. Na Figura 4.17 é apresentado que existem 3 padrões que podem complementar a especificação do requisito Incluir Informação de Filme.

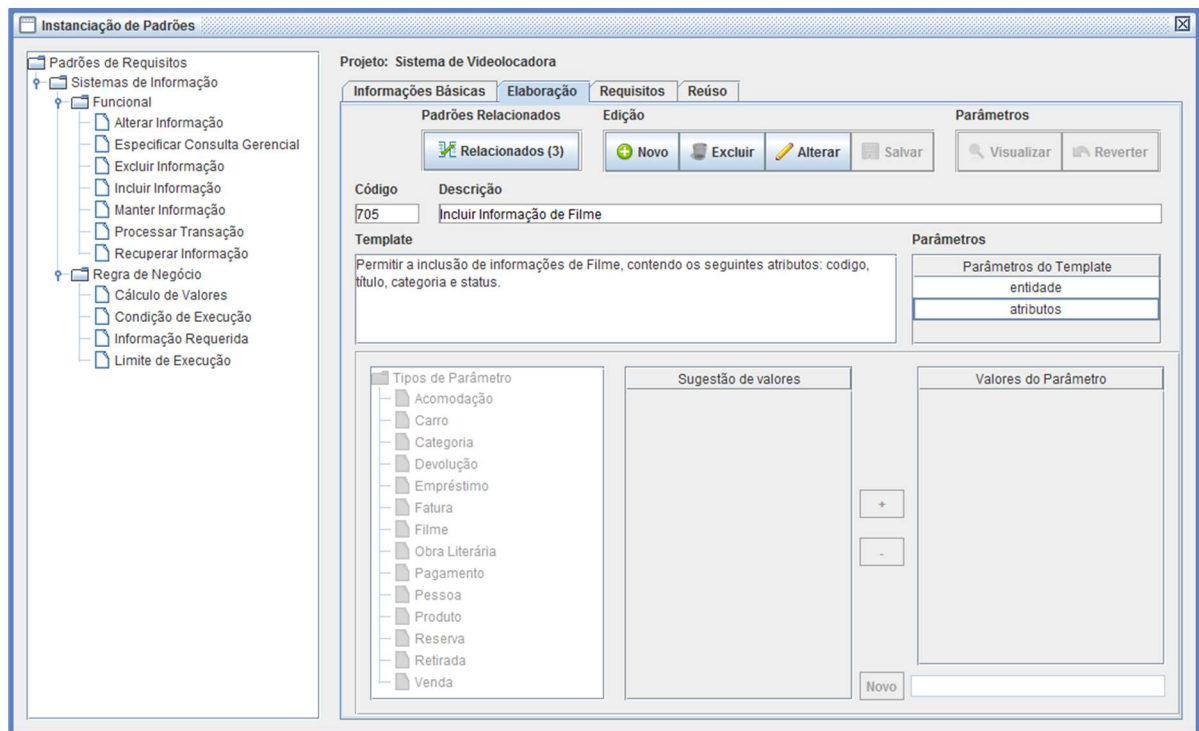


Figura 4.17 – O Apoio Computacional Sugere o Uso de Padrões Relacionados

Ao selecionar o botão [Relacionados] uma caixa de diálogo (Figura 4.18) com os três padrões relacionados é apresentada. Trata-se dos padrões Alterar Informação, Excluir Informação e Recuperar Informação. Esses padrões foram apresentados tendo em vista que quando existe a necessidade de inclusão de informação, possivelmente os *stakeholders* também desejam alterar, excluir ou recuperar uma informação.

Com o objetivo de complementar a especificação do projeto de videolocadora, o padrão relacionado Alterar Informação também será instanciado. Nesse caso, o botão [Instanciar] da caixa de diálogo deve ser pressionado.

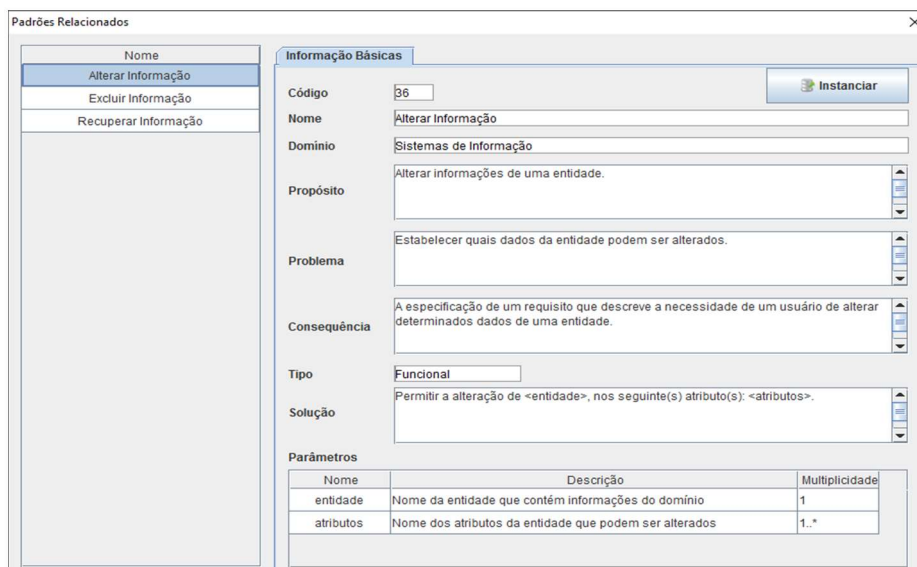


Figura 4.18 – Caixa de Diálogo com os Padrões Relacionados ao Padrão Incluir Informação

Na Figura 4.19 é apresentada a escrita do requisito para a alteração de filme. Entretanto, por ter sido instanciado a partir de outro padrão, para facilitar o preenchimento dos parâmetros, o apoio computacional sugere os mesmos valores utilizados no requisito Incluir Informação de Filme. Isso pode ser observado na Figura 4.19 que ao selecionar o parâmetro *entidade*, a sugestão Filme foi apresentada. O mesmo pode ser observado no parâmetro *atributos* na Figura 4.20, no entanto nesse parâmetro foram atribuídos apenas os valores “título” e “categoria” como aqueles que são permitidos para alteração.

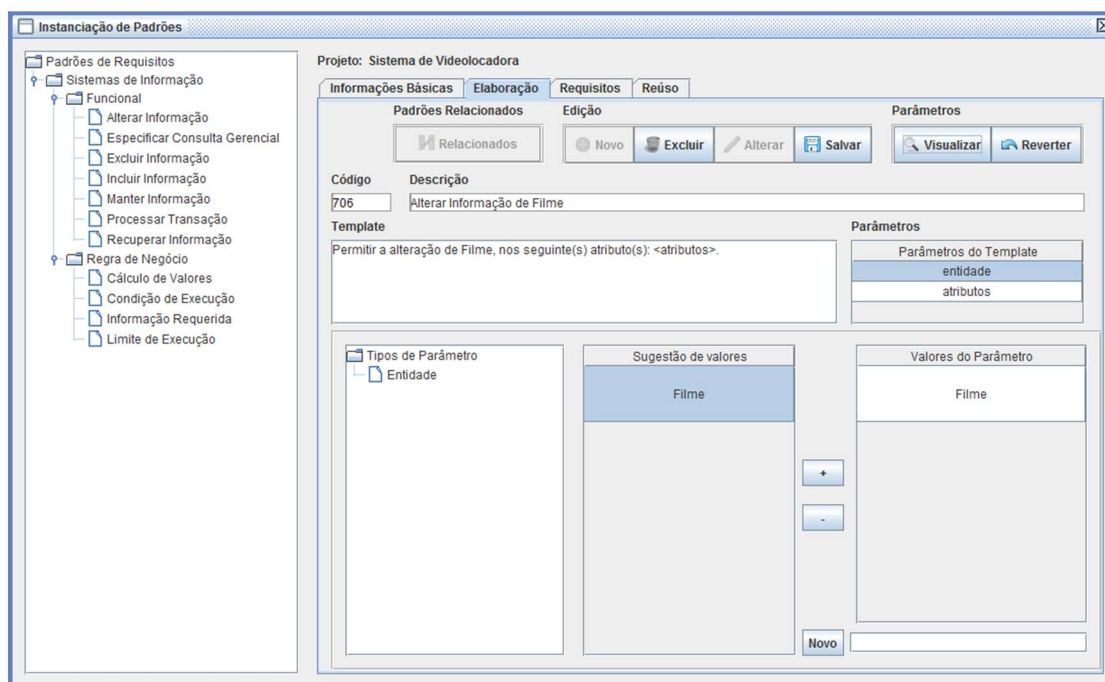


Figura 4.19 – Escrita do Requisito para Alterar Informação de Filme: Preenchimento do Parâmetro *entidade*

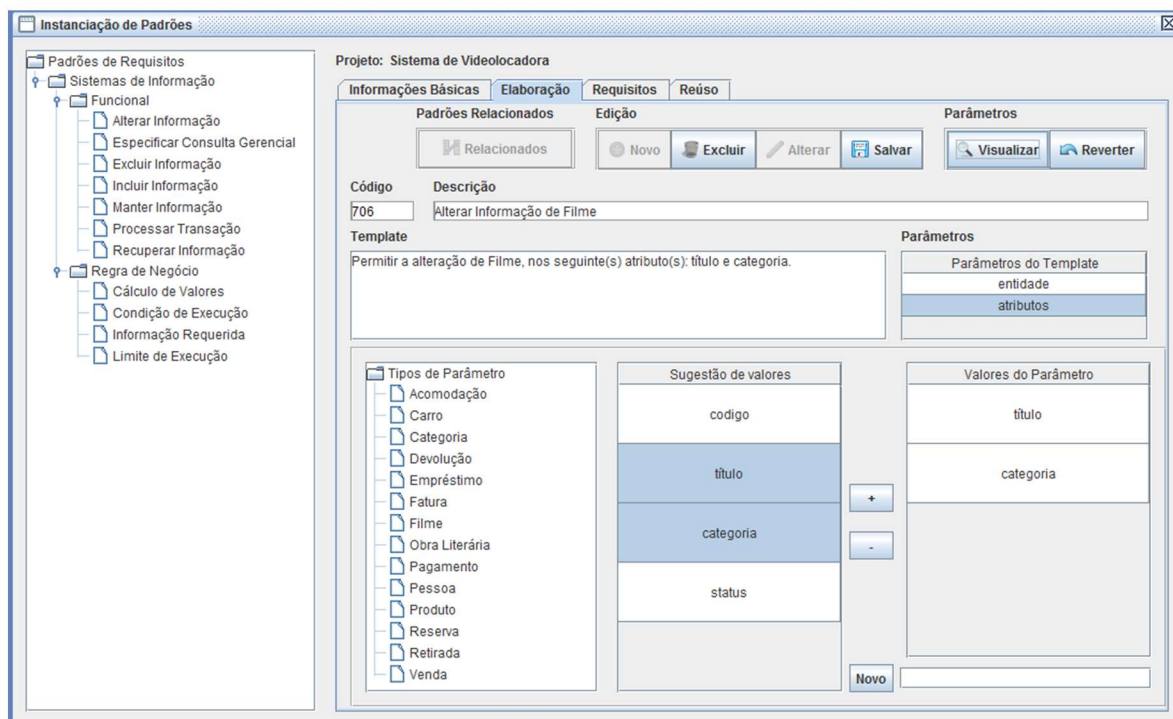


Figura 4.20 – Escrita do Requisito para Alterar Informação de Filme: Preenchimento do Parâmetro *atributos*

A escrita de requisitos para a inclusão de cliente é muito comum em SI, inclusive no repositório do apoio computacional, há requisitos de outros projetos com essa finalidade. Nesse sentido, ao invés de instanciar o padrão Incluir Informação para a escrita do requisito de inclusão de cliente, é apresentado a reutilização de um requisito a partir do repositório. Na Figura 4.21 é mostrada a Aba *Reúso* apresentando requisitos para a inclusão de informação de diferentes entidades de dados presentes no repositório. O requisito selecionado para reúso foi o de Incluir Informação de Hospede que foi especificado para um projeto de Hotel. No exemplo da Figura 4.21, por meio do botão [*Visualizar*], a descrição do requisito foi apresentada em uma caixa de diálogo, onde é possível constatar que esse requisito atende ao objetivo do sistema de videolocadora. O botão [*Reusar*] deve ser pressionado para a reutilização do requisito.

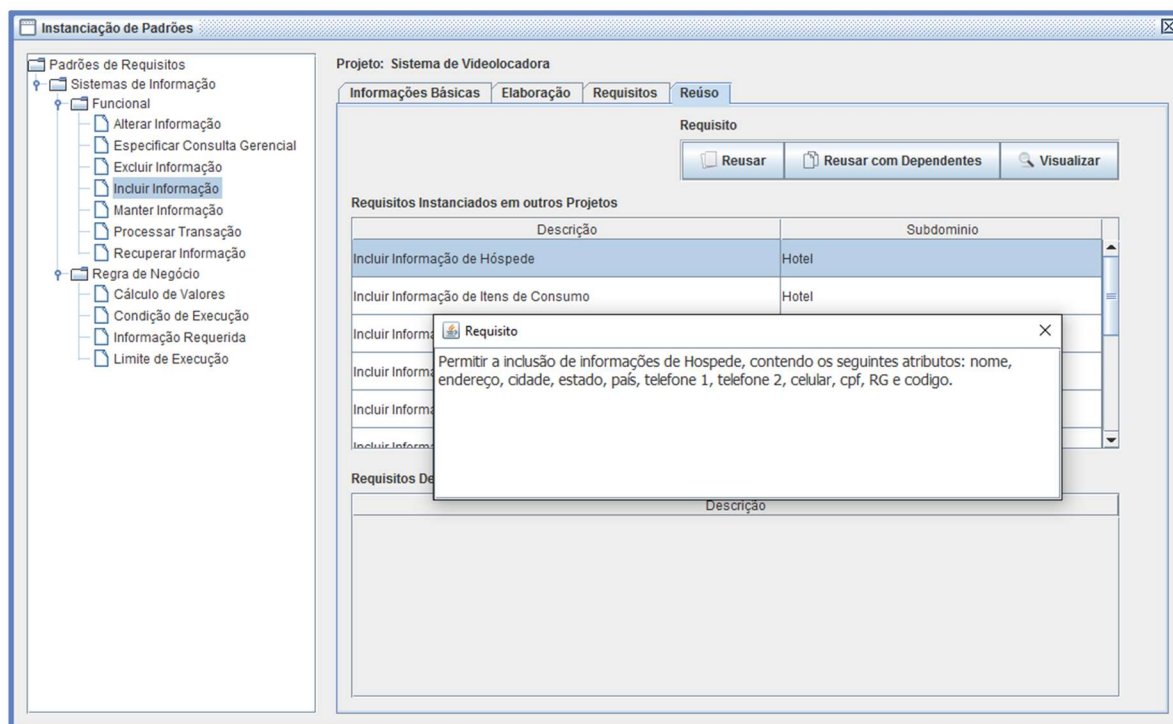


Figura 4.21 – Visualização de Requisitos de Projetos Concluídos

A aba *Requisitos*, mostrada na Figura 4.22, apresenta todos os requisitos do projeto de videolocadora. Para alterar a descrição e a entidade de dados do requisito reusado, o botão [Editar] deve ser selecionado. Ao selecioná-lo é solicitado a confirmação de edição.

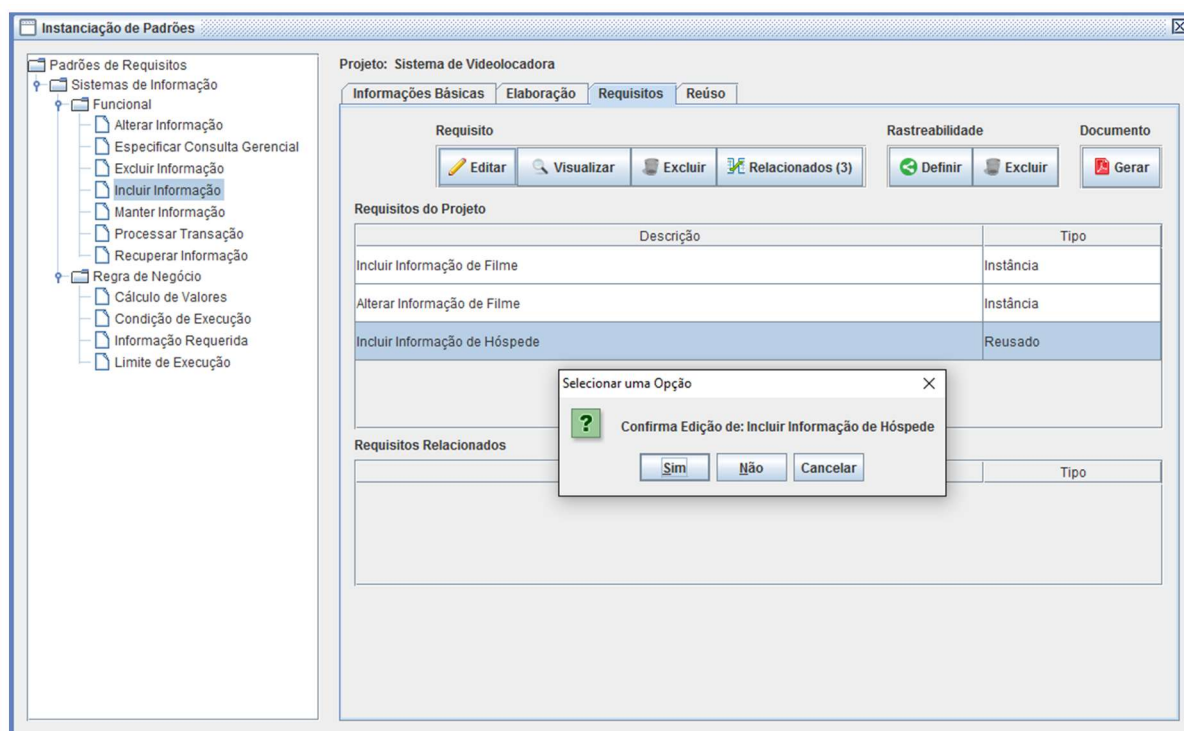


Figura 4.22 – Editando Requisito Reusado de Projetos Concluídos

Na Figura 4.23 é apresentada a alteração do requisito reusado para a inclusão de cliente. Foi alterada a *Descrição* do requisito com a troca da palavra “hospede” por “cliente” e no parâmetro *entidade* há a necessidade de excluir o atual valor por meio do botão [-] e a inclusão do valor cliente, conforme demonstra a atribuição de forma manual. O parâmetro *atributos* não requer alteração em seus valores, tendo em vista serem os mesmos necessários no sistema de videolocadora. Para finalizar o processo, o botão [Salvar] deve ser pressionado. O resultado do requisito salvo, é apresentado na Figura 4.24.

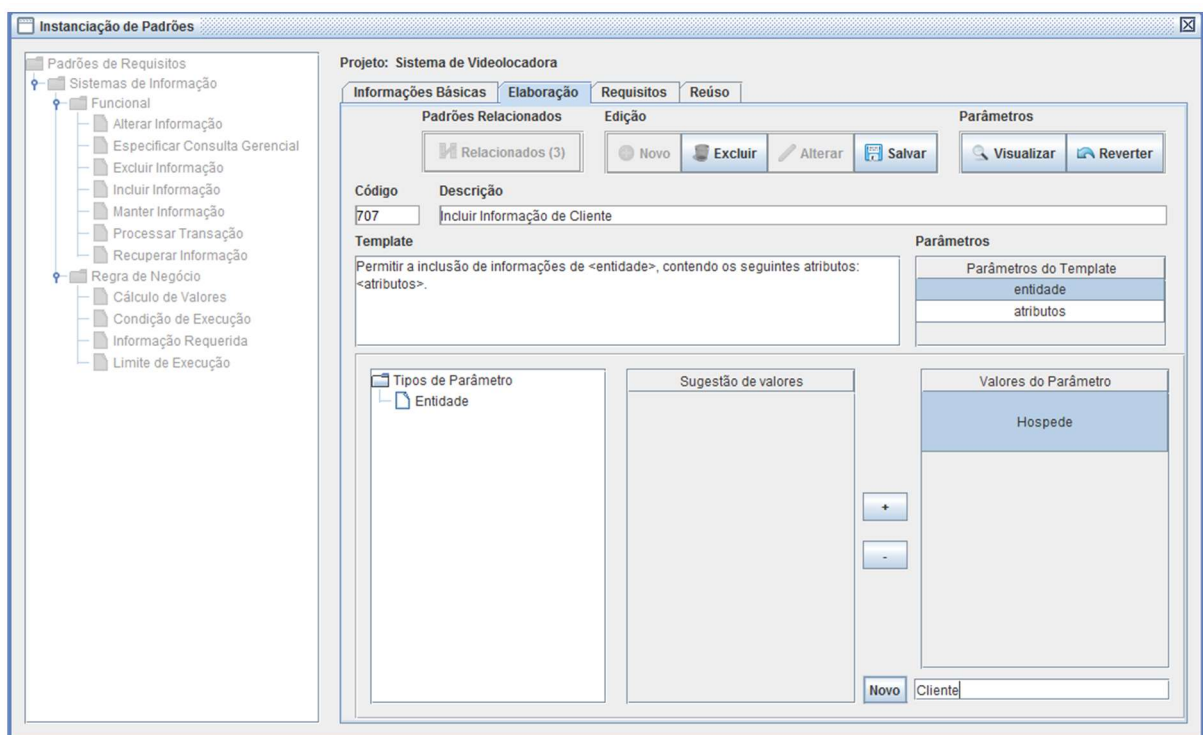


Figura 4.23 – Alteração do Requisito Reusado de Projetos Concluídos

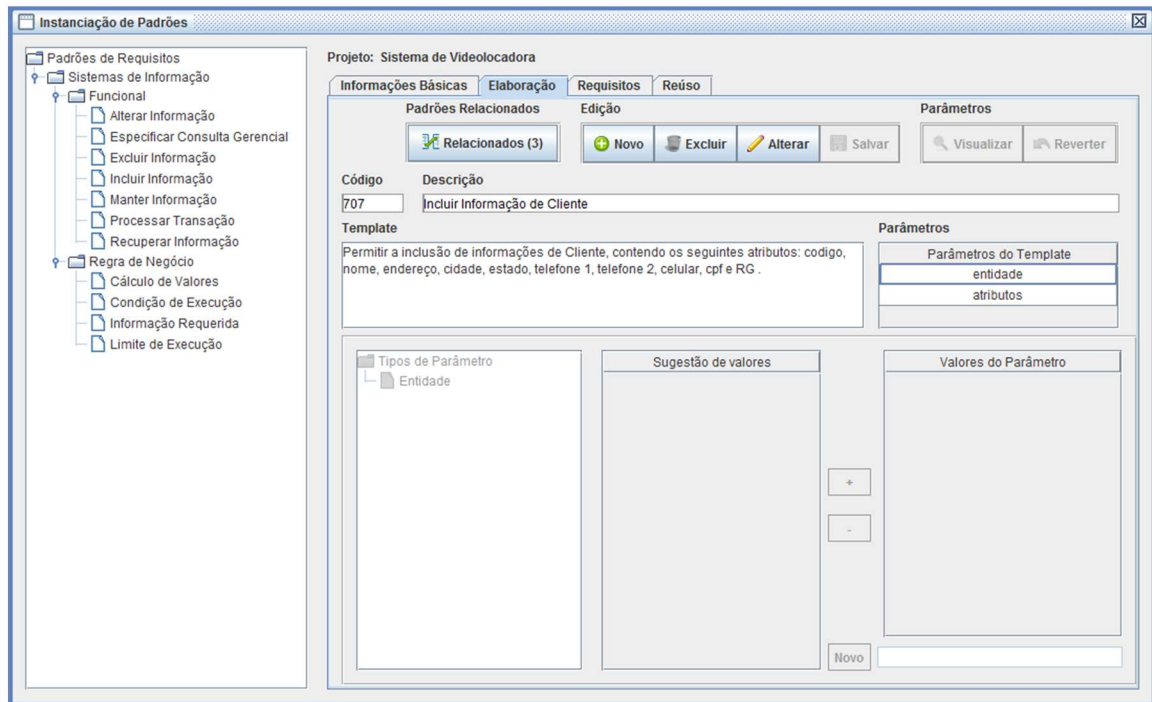


Figura 4.24 – Requisito Reusado para a Inclusão de Cliente

Conforme apresentado durante a instanciação do requisito para a inclusão de filme, o requisito reusado (Figura 4.24) por ter sua origem o padrão Incluir Informação, também apresenta três padrões relacionados. Da mesma forma que o padrão relacionado Alterar Informação foi instanciado, também foi para a alteração de cliente, conforme é mostrado na Figura 4.25.

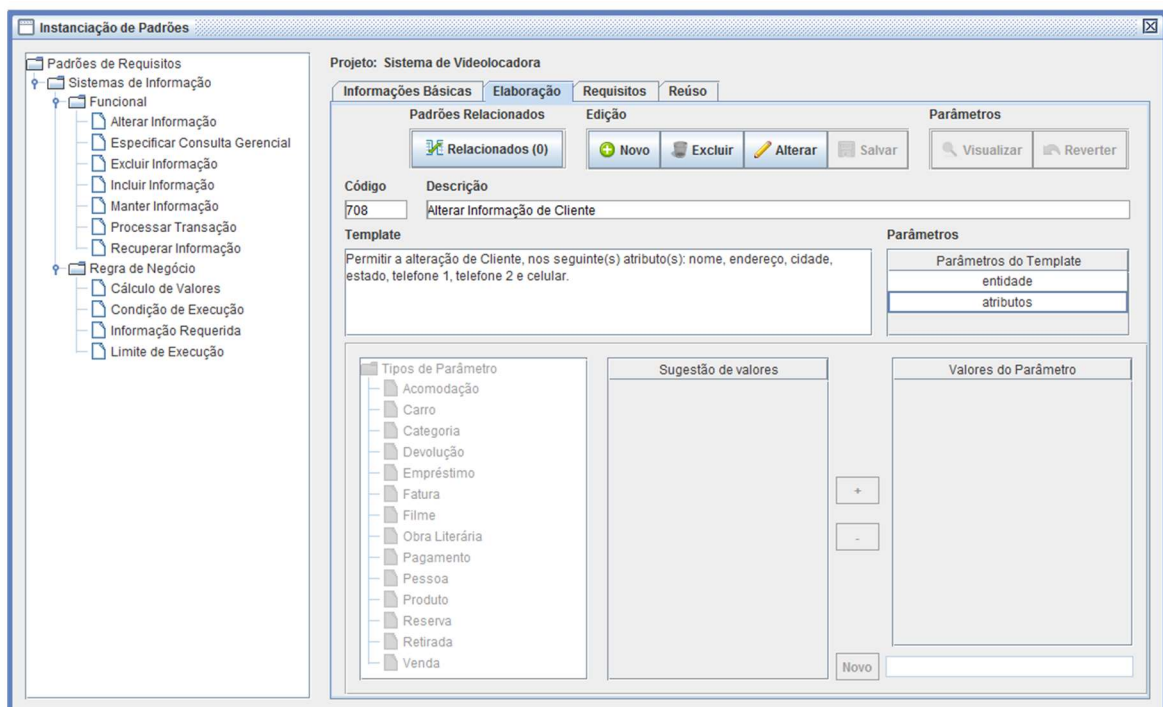


Figura 4.25 – Requisito para Alterar Informação de Cliente

A próxima necessidade descrita na visão geral é: “O sistema deve informar a existência do filme e se está disponível”. Para resolver esse problema, o padrão Especificar Consulta Gerencial foi instanciado conforme exemplos anteriormente apresentados. O resultado da instanciação gerou o requisito Consultar Filme apresentado na Figura 4.26.

O padrão Especificar Consulta Gerencial possui quatro parâmetros, sendo: *operação*, *entidade*, *condição* e *atributos*. Com exceção do parâmetro *condição*, todos os demais foram preenchidos com sugestões do apoio computacional. O parâmetro *operação* foi preenchido com o valor “consulta”, o parâmetro *entidade* com o valor “Filme”, o parâmetro *condição* com o valor “pelo título do filme” e o parâmetro *atributos* com os valores “título” e “status”.

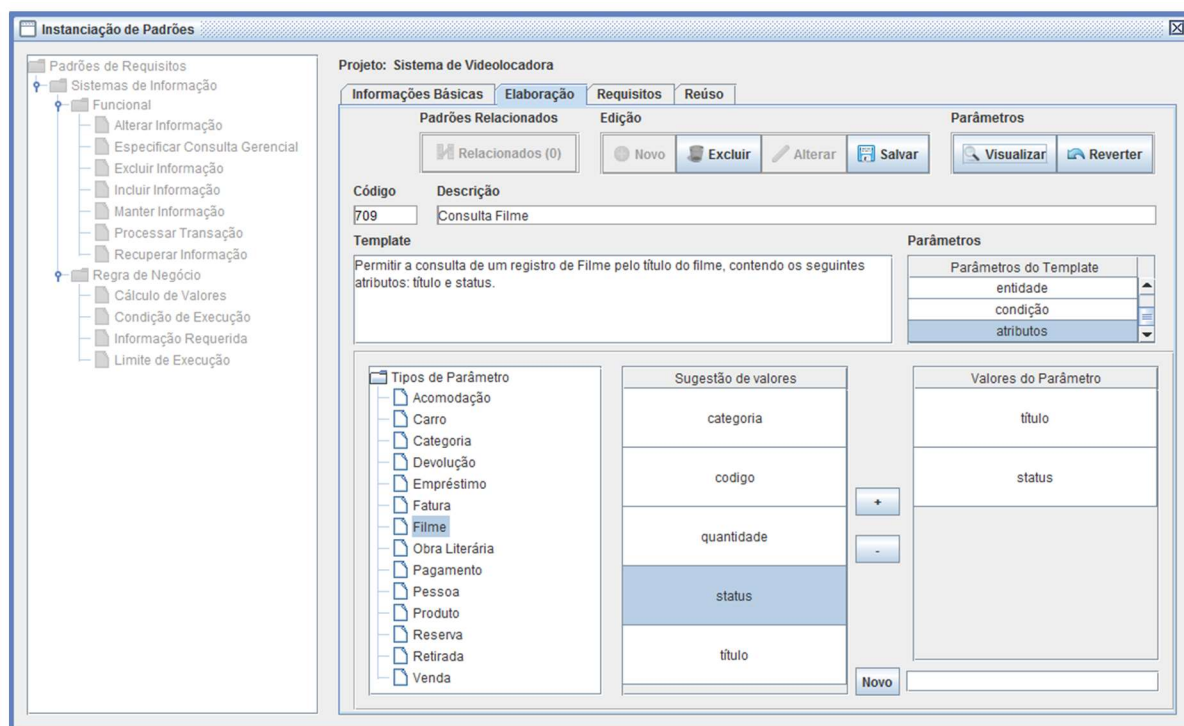


Figura 4.26 – Requisito para a Consulta da Situação do Filme

Outra necessidade descrita na visão geral é: “Se o cliente quiser alugar o filme, o empréstimo é registrado no sistema”. Para atender a esse objetivo o padrão Processar Transação foi instanciado. As próximas figuras mostram os parâmetros sendo preenchidos. Na Figura 4.27 é apresentado o parâmetro *transação* sendo preenchido com o valor “Empréstimo”, na Figura 4.28 o parâmetro *entidade* foi preenchido com o valor “Filme” e na Figura 4.29 o parâmetro *atributos* foi preenchido com os valores “código”, “identificação do cliente”, “identificação do filme”, “data do

empréstimo”, “data da devolução” e “valor”. Todos os parâmetros foram preenchidos usando as sugestões do apoio computacional.

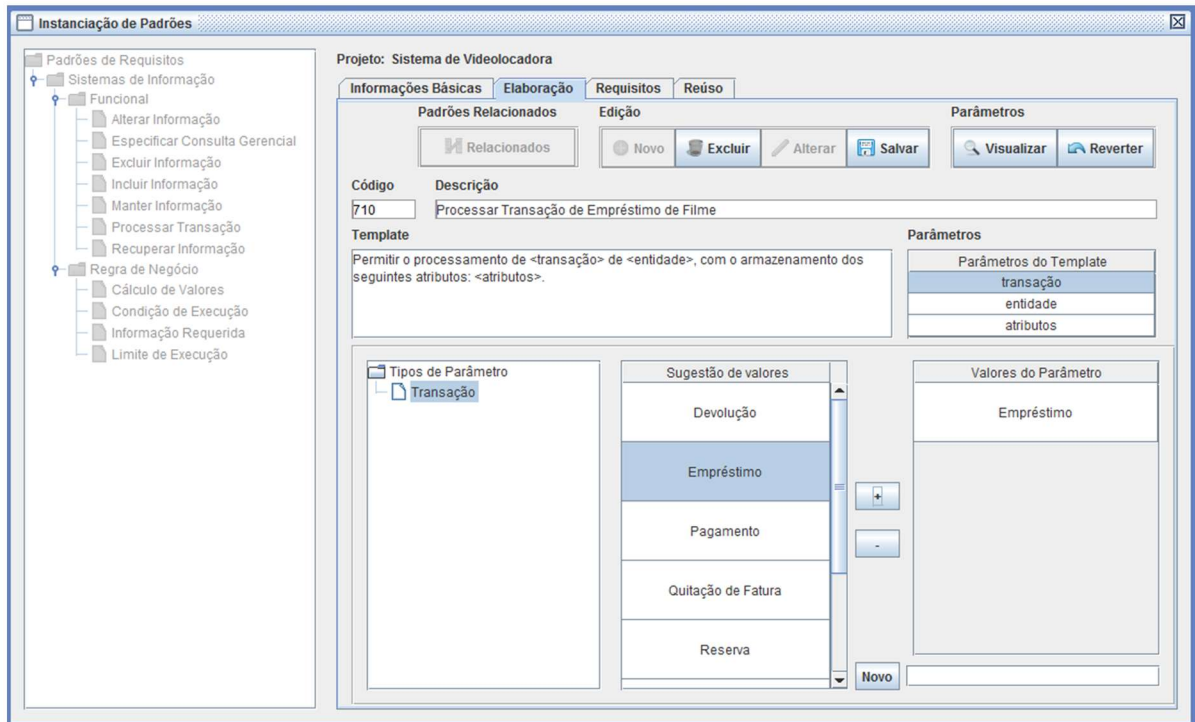


Figura 4.27 – Escrita do Requisito Processar Transação de Empréstimo de Filme: Preenchimento do Parâmetro *transação*

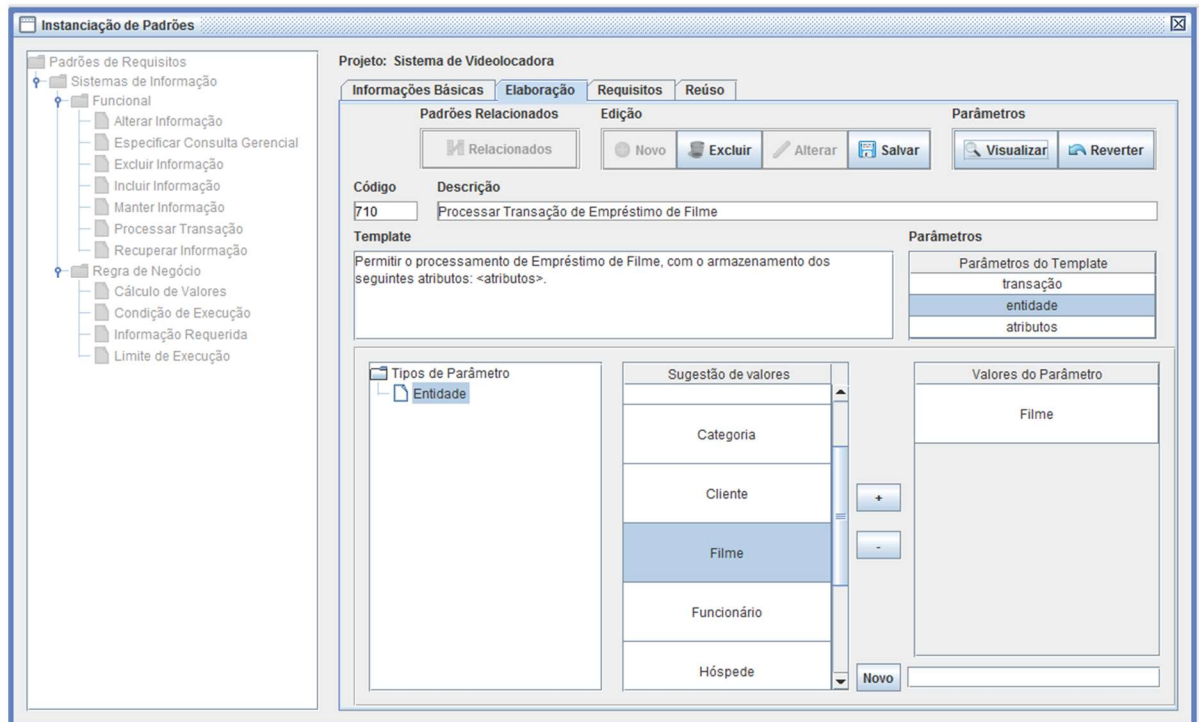


Figura 4.28 – Escrita do Requisito Processar Transação de Empréstimo de Filme: Preenchimento do Parâmetro *entidade*

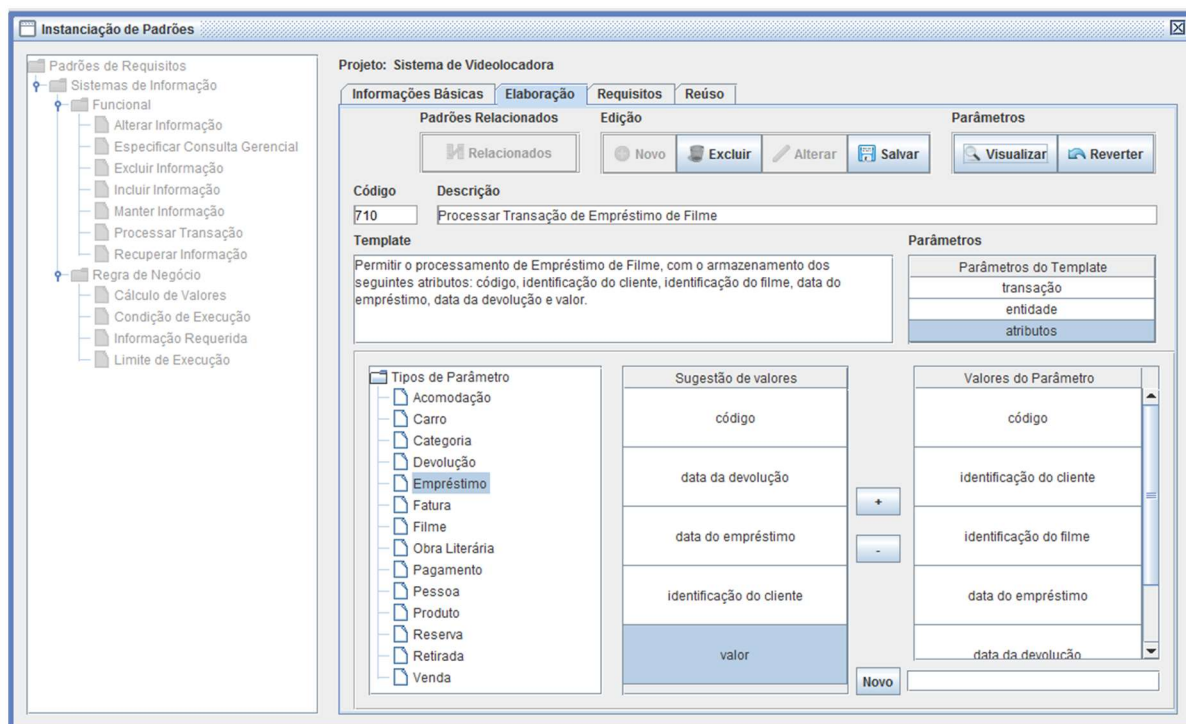


Figura 4.29 – Escrita do Requisito Processar Transação de Empréstimo de Filme: Preenchimento do Parâmetro atributos

Na Figura 4.30 é ilustrado que o requisito escrito para o processamento de empréstimo de filme possui cinco padrões relacionados para complementar a especificação do requisito. Geralmente, requisitos que tratam o processamento de transações requerem a definição de regras de negócio para o estabelecimento das condições de operação. Esse fato é evidenciado na visão geral de exemplo, onde relata que “O empréstimo é realizado somente se o filme estiver disponível”. Nesse sentido, o apoio computacional fornece como sugestão a utilização de quatro padrões relacionados para a escrita de regras de negócio e um relacionado à manutenção de dados, conforme é apresentado na Figura 4.31.

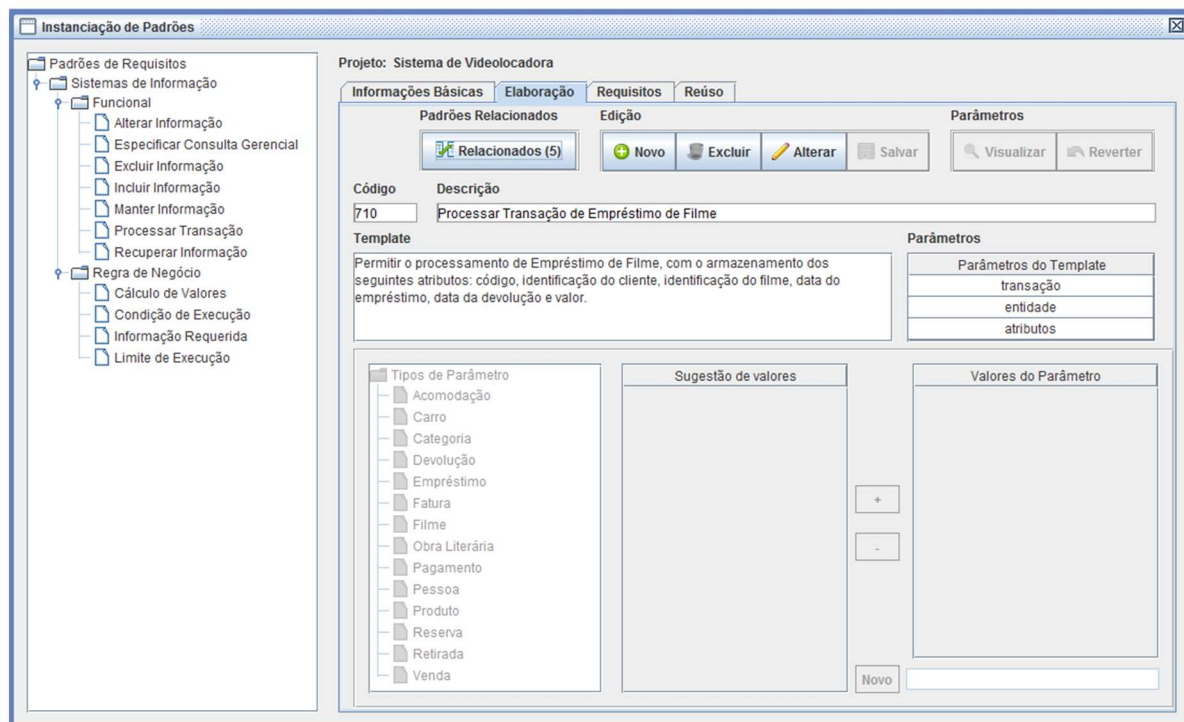


Figura 4.30 – Requisito para o Processamento de Empréstimo de Filme

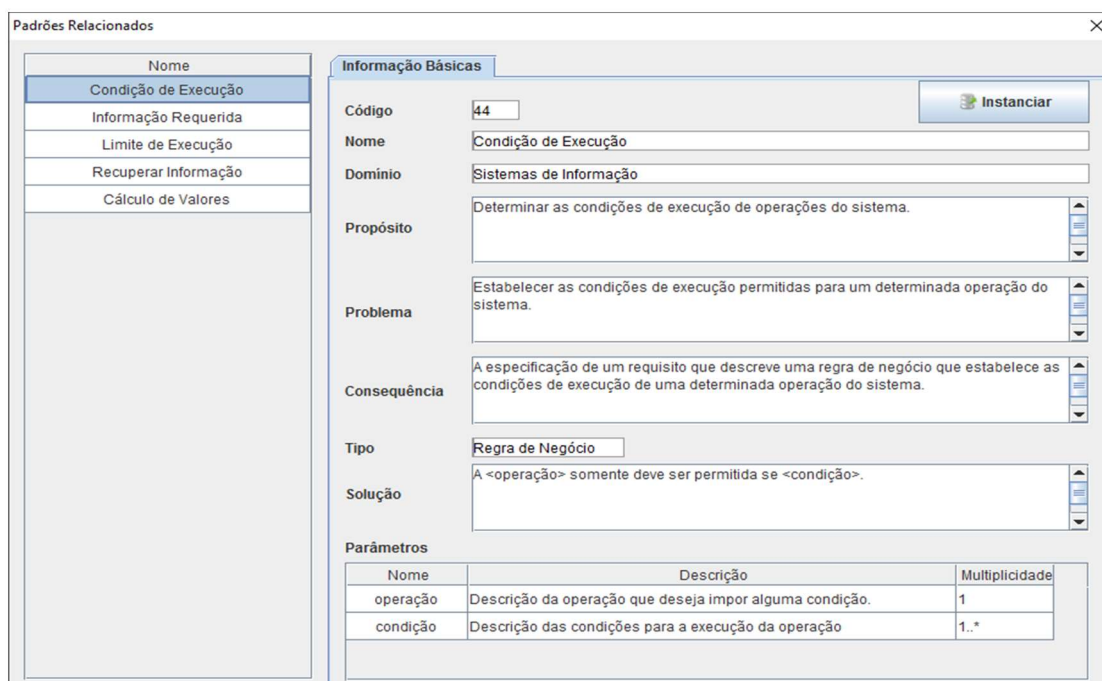


Figura 4.31 – Caixa de Diálogo com os Padrões Relacionados ao Padrão Processar Transação

Nas Figuras 4.32 e 4.33 é apresentada a instanciação da regra de negócio Condição de Execução para a escrita do requisito Condição de Empréstimo. O parâmetro *operação* foi preenchido com o valor “operação de empréstimo” e o parâmetro *condição* com o valor “o filme estiver disponível”.

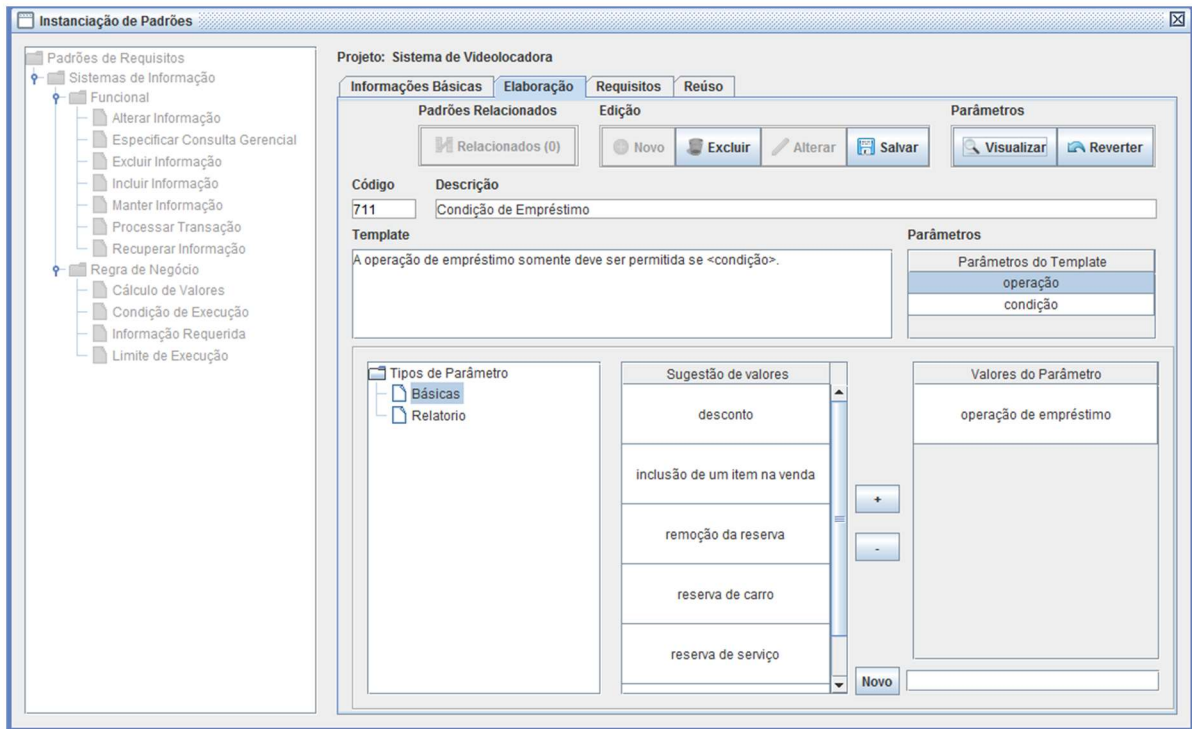


Figura 4.32 – Escrita do Requisito Condição de Empréstimo: Preenchimento do Parâmetro *operação*

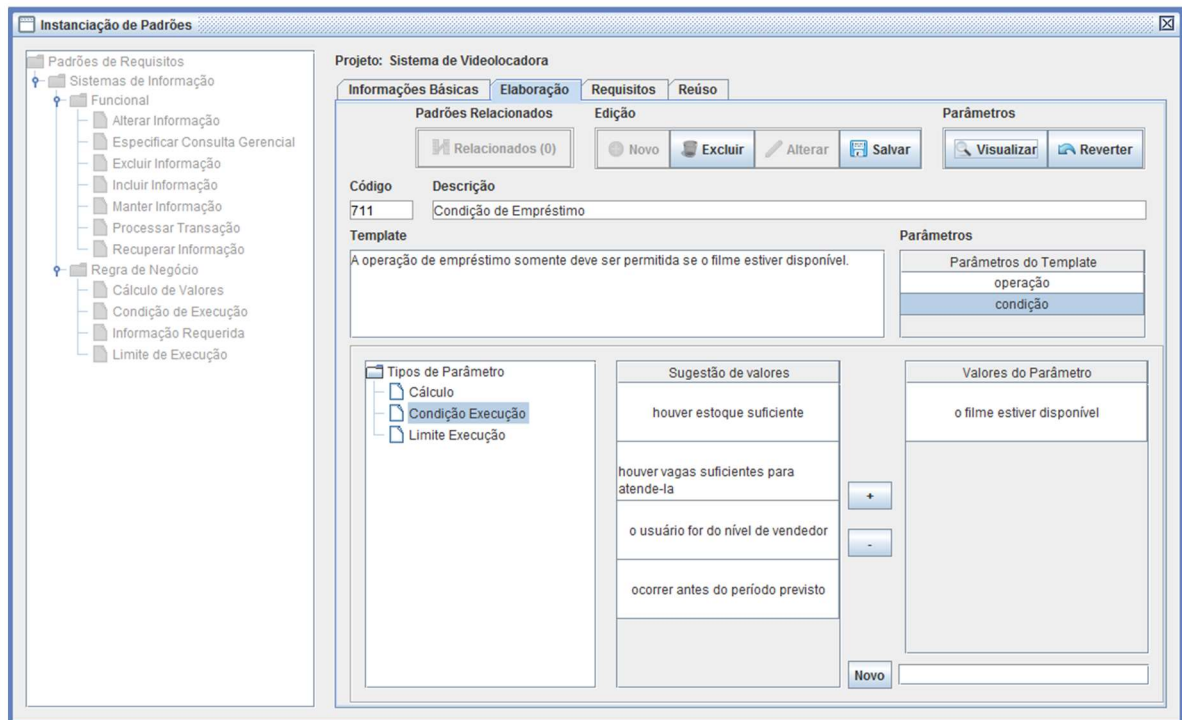


Figura 4.33 – Escrita do Requisito Condição de Empréstimo: Preenchimento do Parâmetro *condição*

Ainda relacionado ao empréstimo de filmes, é relatado na visão geral que: “Para a efetivação do empréstimo do filme reservado, o sistema deve ser capaz de recuperar a reserva previamente realizada”. Para atender a esse propósito, o padrão relacionado

Recuperar Informação foi instanciado e apresentado nas Figuras 4.34, 4.35 e 4.36. O parâmetro *entidade* foi preenchido com o valor “reserva”, o parâmetro *atributos* com o valor “identificação do cliente” e o parâmetro *condição* com o valor “durante o processamento do empréstimo”.

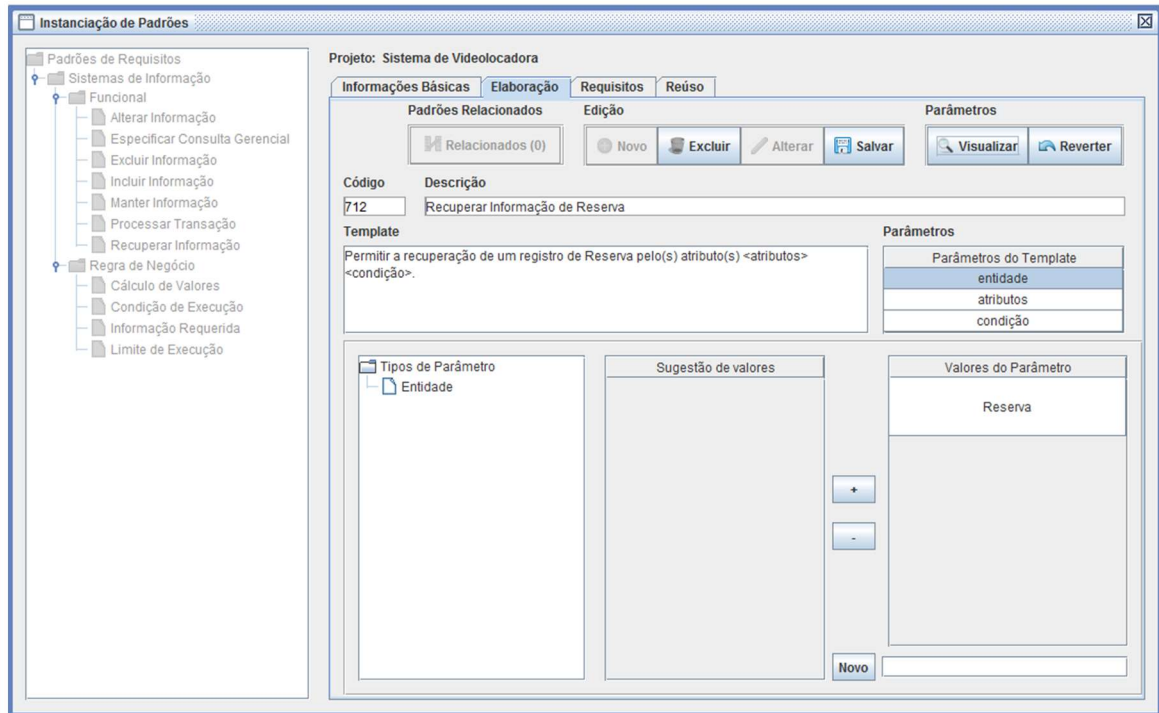


Figura 4.34 – Escrita do Requisito Recuperar Informação de Reserva: Preenchimento do Parâmetro *entidade*

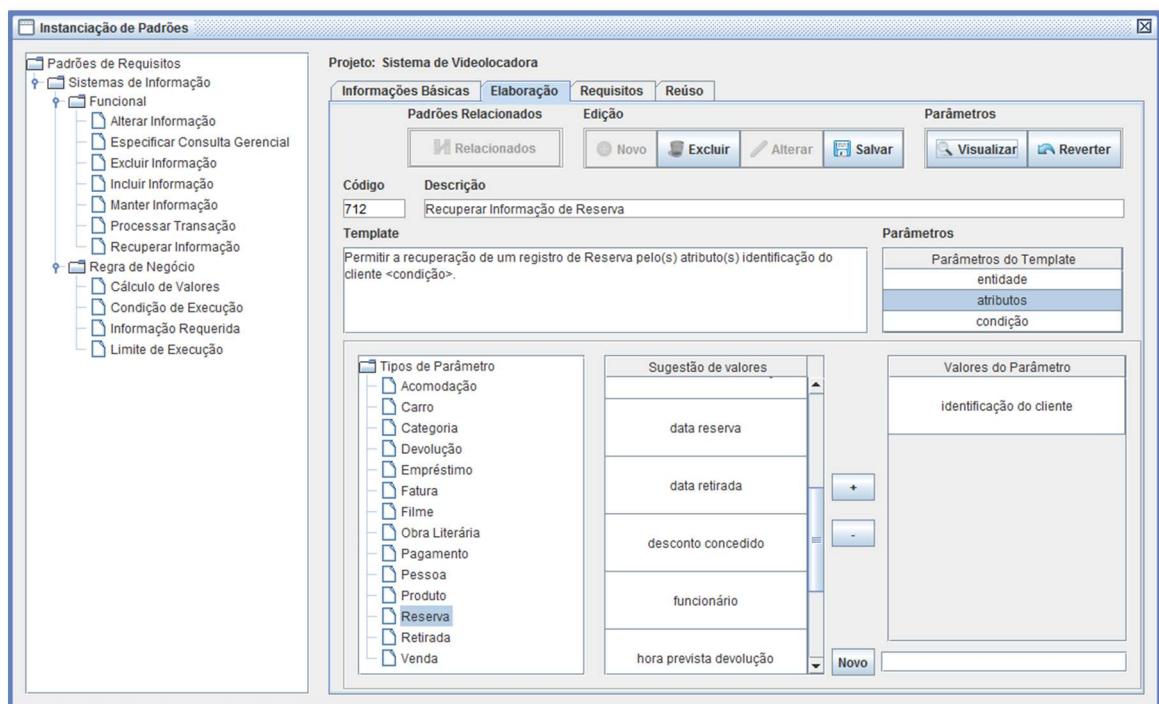


Figura 4.35 – Escrita do Requisito Recuperar Informação de Reserva: Preenchimento do Parâmetro *atributos*

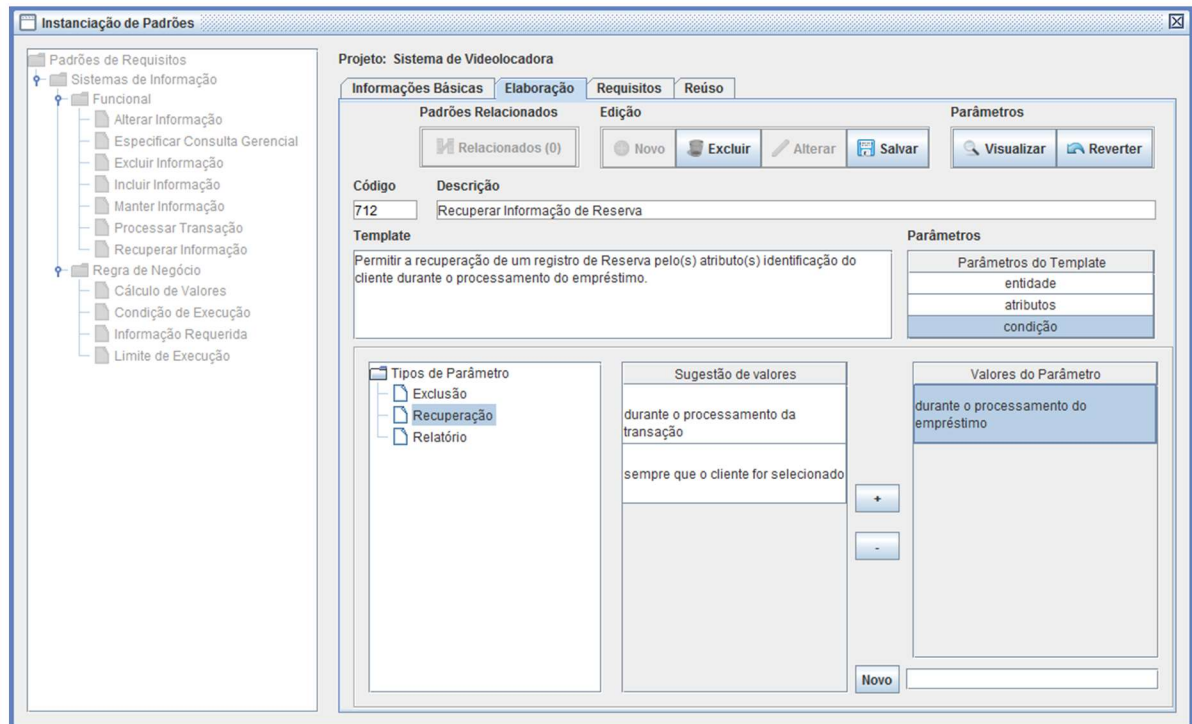


Figura 4.36 – Escrita do Requisito Recuperar Informação de Reserva: Preenchimento do Parâmetro *condição*

Na aba *Requisitos* além de apresentar os requisitos especificados, também apresenta os requisitos relacionados. Na Figura 4.37 o requisito *Processar Transação de Empréstimo de Filme* está selecionado e possui dois requisitos relacionados: *Condição de Empréstimo* e *Recuperar Informação de Reserva*.

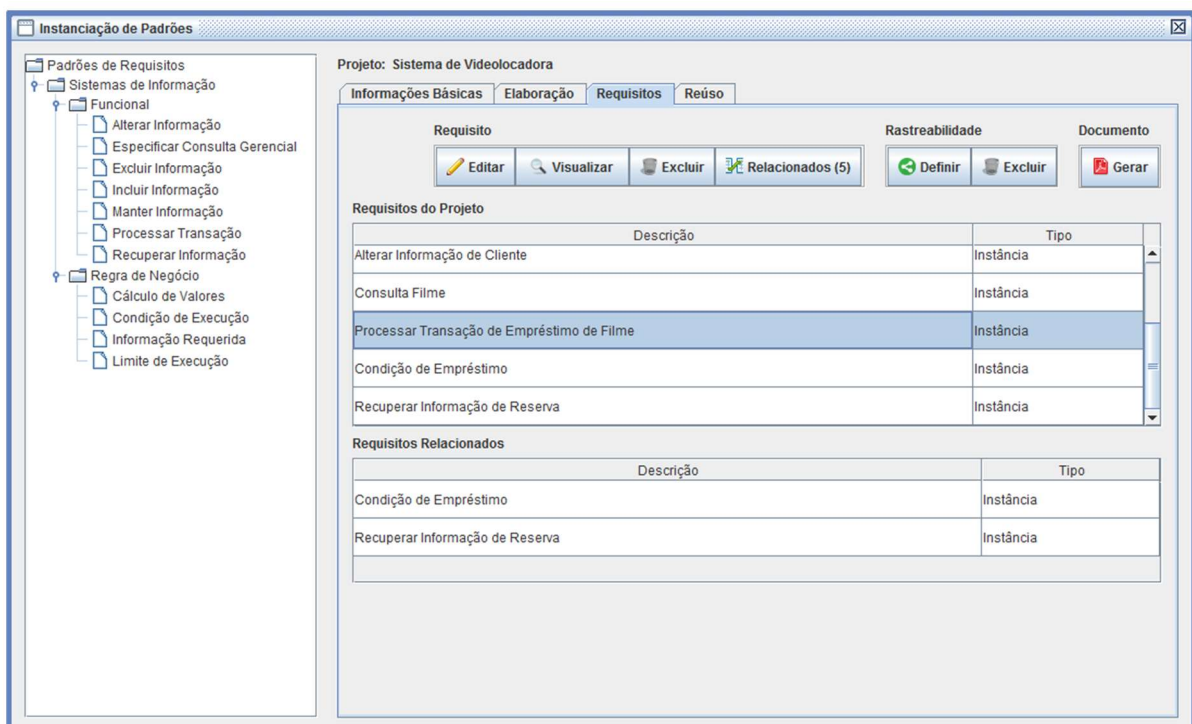


Figura 4.37 – Requisitos do Projeto de Videolocadora

A última necessidade relatada na visão geral de exemplo é “Se o filme não estiver disponível, o cliente poderá efetuar uma reserva”. Para atender a esse propósito, o padrão Processar Transação foi instanciado e apresentado nas Figuras 4.38, 4.39 e 4.40. O parâmetro *transação* foi preenchido com o valor “reserva”, o parâmetro *entidade* com o valor “filme” e o parâmetro *atributos* com os valores “código”, “identificação do cliente”, “identificação do filme”, “data reserva” e “data retirada”.

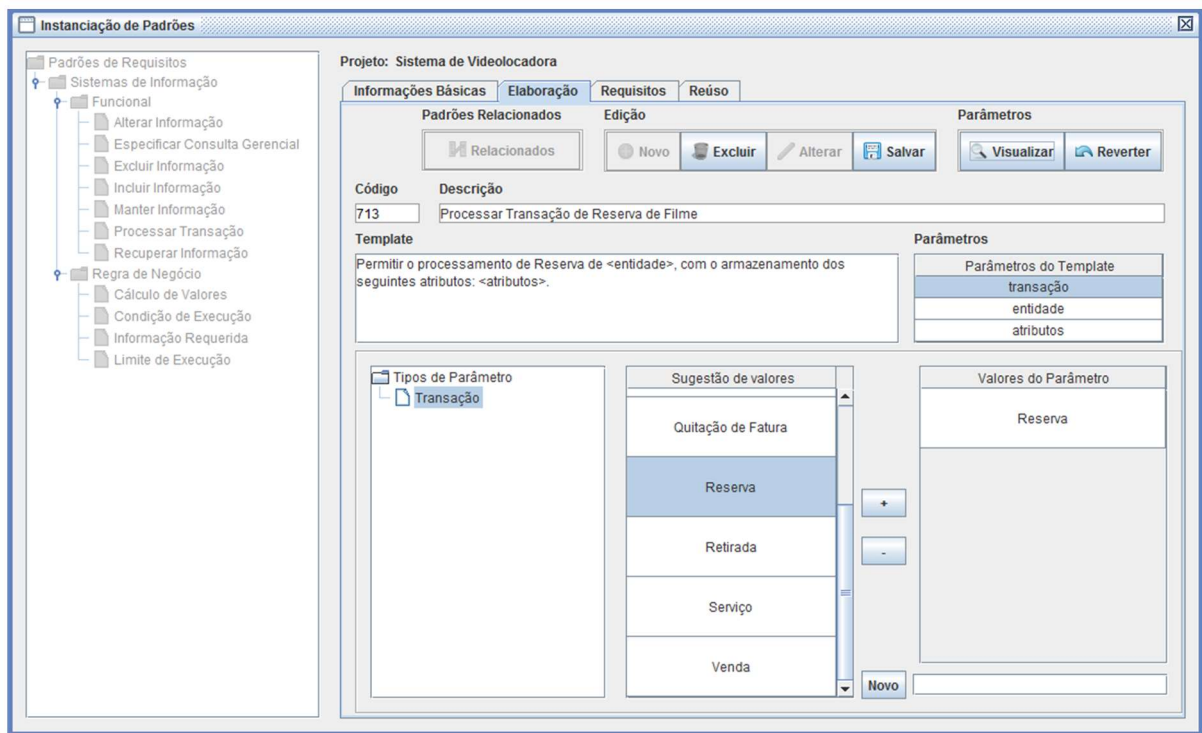


Figura 4.38 – Escrita do Requisito Processar Transação de Reserva: Preenchimento do Parâmetro *transação*

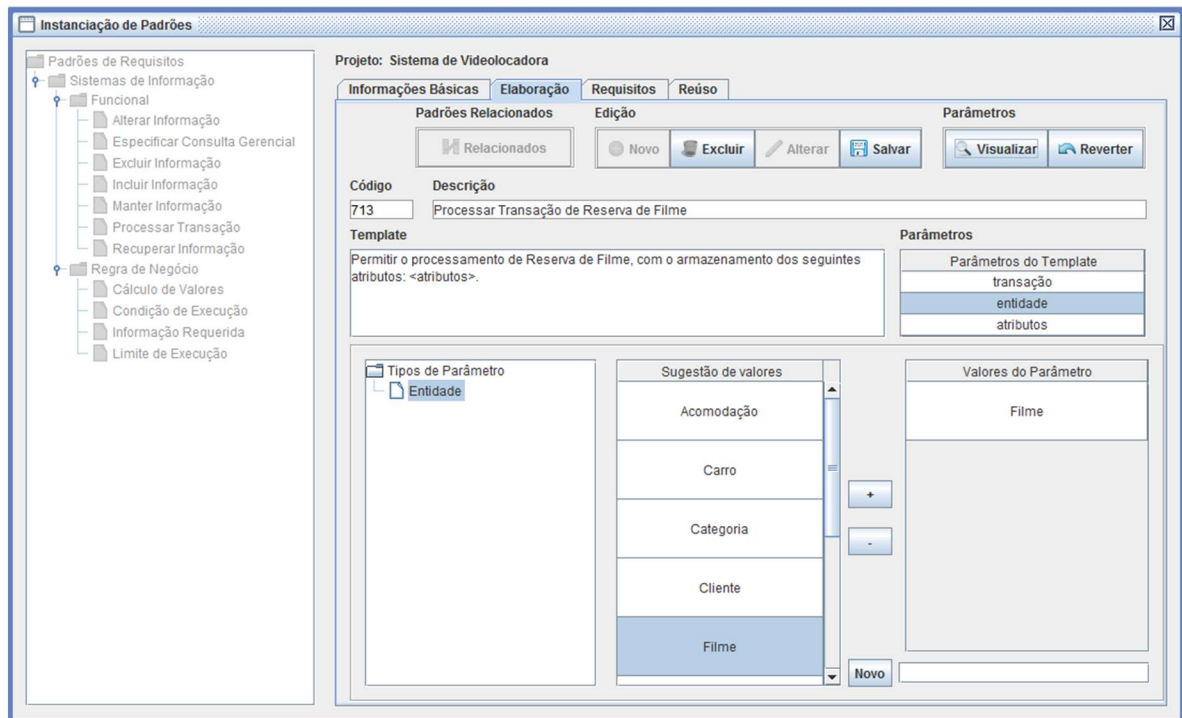


Figura 4.39 – Escrita do Requisito Processar Transação de Reserva: Preenchimento do Parâmetro *entidade*

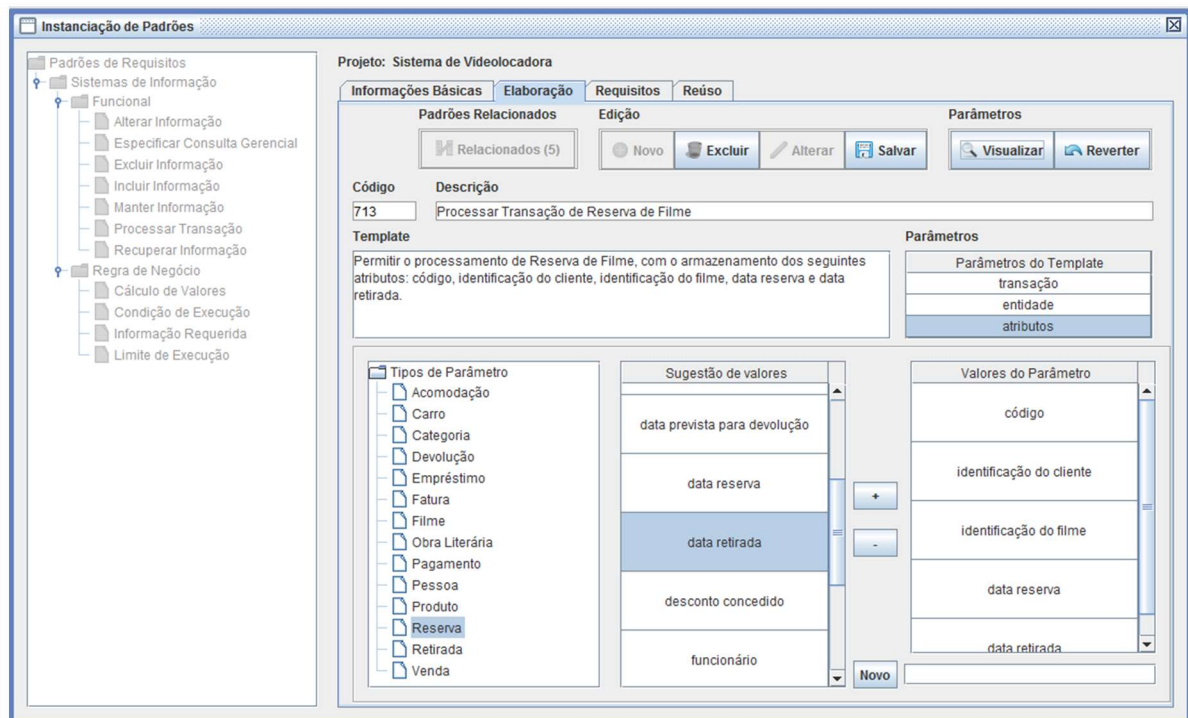


Figura 4.40 – Escrita do Requisito Processar Transação de Reserva: Preenchimento do Parâmetro *atributos*

Para complementar a especificação do sistema de videolocadora, um requisito com o objetivo de imprimir os empréstimos de filmes da data atual foi escrito. Esse requisito foi instanciado pelo padrão Especificar Consulta Gerencial, o mesmo

utilizado para a Consulta da Situação do Filme (Figura 4.26). O resultado da instanciação é apresentado na Figura 4.41, sendo que: o parâmetro *operação* foi preenchido com o valor “impressão”, o parâmetro *tipo* com o valor “um relatório”, o parâmetro *entidade* com o valor “Filmes”, o parâmetro *condição* com o valor “emprestados na data atual” e o parâmetro *atributos* com os valores “nome do cliente” e “título do filme”.

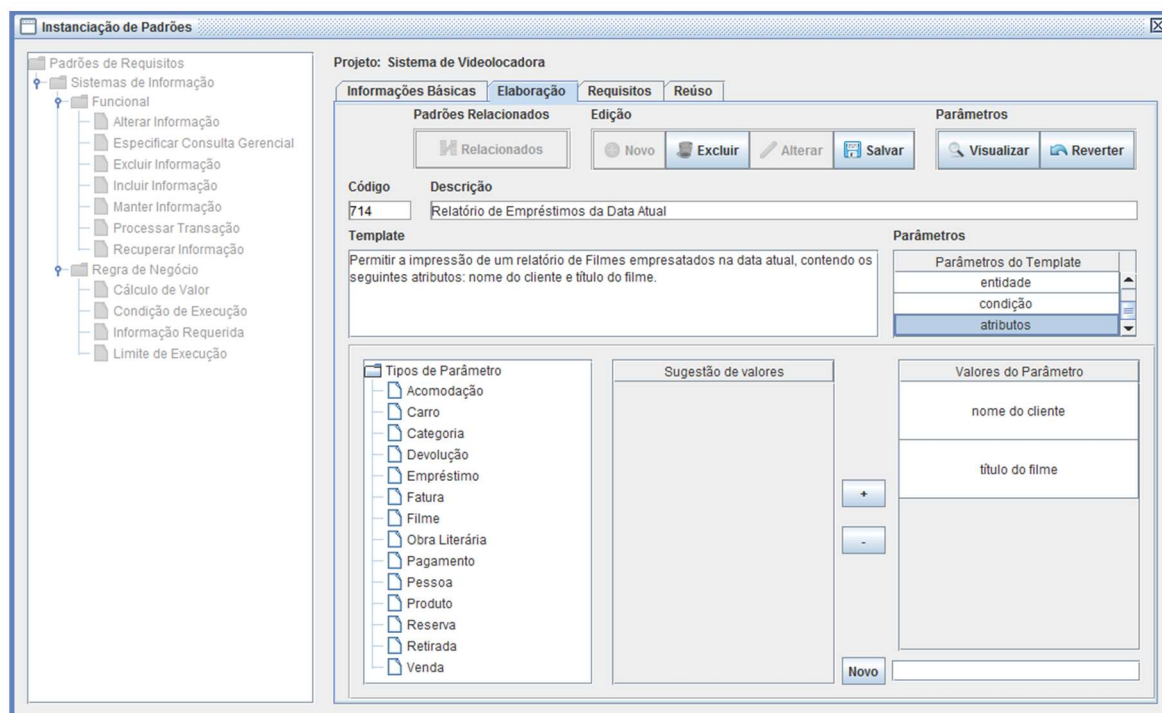


Figura 4.41 – Requisito para a Impressão de Relatório de Empréstimos de Filme da Data Atual

Após a escrita de requisitos, o botão [*Gerar*] na aba *Requisitos* pode ser pressionado para a geração do DR que é apresentado em tela e gravado no formato *PDF* na pasta “C:\SRS”. A criação da pasta é realizada automaticamente na primeira vez que o botão [*Gerar*] for pressionado. O DR resultante da instanciação dos padrões apresentados nesta seção é mostrado na Figura 4.42. Os requisitos instanciados por meio de padrões relacionados foram obtidos de forma automática a partir da relação de dependência existente no apoio computacional.

Sistema de Videolocadora
Documento de Requisitos

Visão Geral

O cliente vai até a videolocadora para alugar um filme. Ao chegar no balcão, informa o título do filme para consulta. O sistema deve informar a existência do filme e se está disponível. Se o cliente quiser alugar o filme, o empréstimo é registrado no sistema. O empréstimo é realizado somente se o filme estiver disponível. Se o filme não estiver disponível, o cliente poderá efetuar uma reserva. Para a efetivação do empréstimo do filme reservado, o sistema deve ser capaz de recuperar a reserva previamente realizada.

Requisitos Funcionais

Identificador:	Descrição	Depende de:
RF1	Permitir a inclusão de informações de Filme, contendo os seguintes atributos: código, título, categoria e status.	
RF2	Permitir a alteração de Filme, nos seguinte(s) atributo(s): título e categoria.	RF1.
RF3	Permitir a inclusão de informações de Cliente, contendo os seguintes atributos: código, nome, endereço, cidade, estado, telefone 1, telefone 2, celular, cpf e RG.	
RF4	Permitir a alteração de Cliente, nos seguinte(s) atributo(s): nome, endereço, cidade, estado, telefone 1, telefone 2 e celular.	RF3.
RF5	Permitir a consulta de um registro de Filme pelo título do filme, contendo os seguintes atributos: título e status.	
RF6	Permitir o processamento de Empréstimo de Filme, com o armazenamento dos seguintes atributos: código, identificação do cliente, identificação do filme, data do empréstimo, data da devolução e valor. RN6.1. A operação de empréstimo somente deve ser permitida se o filme estiver disponível.	RF7.
RF7	Permitir a recuperação de um registro de Reserva pelo(s) atributo(s) identificação do cliente durante o processamento do empréstimo.	
RF8	Permitir o processamento de Reserva de Filme, com o armazenamento dos seguintes atributos: código, identificação do cliente, identificação do filme, data reserva e data retirada.	
RF9	Permitir a impressão de um relatório de Filmes emprestados na data atual, contendo os seguintes atributos: nome do cliente e título do filme.	

Figura 4.42 – Documento de Requisitos Elaborado com o Apoio Computacional

4.6 Considerações Finais

Neste capítulo foram apresentadas as funções existentes no apoio computacional para a especificação e gestão dos padrões, a instanciação dos padrões durante a elicitação de requisitos e a manutenção de usuário, cliente e projeto, bem como um exemplo de aplicação.

O apoio computacional fornece orientações ao engenheiro de software para a identificação e escrita de requisitos com a utilização dos padrões e elaboração de um DR com os requisitos funcionais e regras de negócio. Os requisitos não funcionais estão fora do escopo deste projeto de mestrado. As funções implementadas no apoio computacional visaram atender a facilidade, a rapidez e a qualidade na escrita de requisitos, principalmente para desenvolvedores com pouca experiência.

A elaboração do DR a partir do apoio computacional, proporciona os seguintes benefícios para a escrita dos requisitos por meio dos padrões:

- Fornecimento de sugestões de preenchimento para os parâmetros do *template* da solução;
- A reutilização de requisitos especificados em projetos concluídos;
- O fornecimento de sugestões de padrões relacionados;
- A definição de dependência entre os requisitos relacionados.

O apoio computacional visa facilitar: a completude no DR, uma vez que pode fornecer padrões relacionados para complementar a especificação; a agilidade, porque o requisito não precisa ser escrito a partir do zero; a reutilização de requisitos de outros projetos; e a consistência na especificação, tendo em vista que os requisitos foram escritos usando padrões que foram elaborados a partir de experiências de engenheiros de software e projetos de sucesso. Assim, infere-se que o apoio computacional fornece todo o subsídio para a elaboração de um DR completo e consistente e para o aumento da produtividade na especificação de requisitos.

No próximo capítulo são apresentados os resultados dos estudos de caso de aplicação do apoio computacional e padrões na academia.

Capítulo 5

ESTUDOS DE CASO

5.1 Considerações Iniciais

Para avaliar a eficácia da abordagem proposta quanto à completude dos padrões e usabilidade do apoio computacional, foram realizados quatro estudos de casos com estudantes de graduação da Universidade Federal de São Carlos (UFSCar), câmpus de São Carlos/SP e da Universidade do Estado de Minas Gerais (UEMG), unidade de Frutal/MG, em disciplinas que abordaram a engenharia de requisitos.

Na Seção 5.2 são apresentados o planejamento e os resultados do estudo de caso 1, que contou com a participação de quarenta e dois estudantes do terceiro período do curso de Bacharelado em Ciência da Computação da UFSCar (BCC-UFSCar). Na Seção 5.3 são apresentados os resultados do estudo de caso 2, que contou com a participação de sessenta estudantes do sexto período também do BCC-UFSCar. Na Seção 5.4 são apresentados os resultados do estudo de caso 3, que contou com a participação de 24 estudantes do curso de Engenharia Elétrica da UFSCar (EE-UFSCar). Na Seção 5.5 são apresentados os resultados do estudo de caso 4, que contou com a participação de trinta e quatro estudantes do sexto período do curso de Bacharelado em Sistemas de Informação da UEMG (BSI-UEMG). As considerações finais são apresentadas na Seção 5.6.

5.2 Estudo de Caso 1

No primeiro estudo de caso, participaram quarenta e dois estudantes do curso de BCC-UFSCar, que não tinham tido disciplinas sobre engenharia de requisitos. Esses estudantes foram divididos em oito grupos, sendo que cada grupo teve que desenvolver um projeto, obtenção de requisitos e modelagem de um sistema relacionado a serviços oferecidos pela UFSCar como: biblioteca, restaurante universitário, locação de quadra, inscrição em disciplina, etc. As atividades foram divididas em cinco etapas, sendo:

- 1) Primeiramente, os estudantes elaboraram um documento de requisitos de modo *ad hoc*, relacionados a algum serviço oferecido pela Universidade, envolvendo poucos requisitos. Essa etapa foi após a apresentação dos conceitos de requisitos e técnicas de obtenção dos mesmos, e que tinha como objetivo a elaboração de DR, conforme orientações fornecidas durante a disciplina;
- 2) Os padrões de requisitos e o apoio computacional foram apresentados depois que o DR descrito em 1) já estava elaborado. Para apresentar o uso dos padrões e do apoio computacional foi elaborado um exemplo e esse foi desenvolvido juntamente com os estudantes. Após essa etapa foi realizado um treinamento, utilizando um sistema exemplo com requisitos diferentes dos que constavam dos DR elaborados pelos estudantes e do exemplo dado anteriormente;
- 3) Em seguida foi solicitada a elaboração do DR do sistema em desenvolvimento pelo grupo, agora utilizando o ferramental apresentado. Essa atividade foi realizada pelo grupo fora da sala de aula, sendo que as dúvidas poderiam ser esclarecidas via e-mail;
- 4) Após essa atividade, os estudantes responderam a um questionário (Apêndice A), a fim de identificar a porcentagem de requisitos escritos com o uso do apoio computacional, se o DR ficou mais completo e se o apoio computacional é de fácil utilização;
- 5) Essa atividade realizada não teve atribuição de nota a fim de não prejudicar os resultados.

Os objetivos desse primeiro estudo de caso foram identificar se com o catálogo de padrões de requisitos elaborado os estudantes iriam conseguir reescrever todos os requisitos escritos de modo *ad hoc*, se o DR resultante ficou mais completo do que o inicialmente e se o apoio computacional é fácil de ser usado, entendido e se apresentou problemas.

No questionário, foi elaborada uma questão aberta para os participantes fazerem suas considerações em relação ao DR elaborado, se o mesmo ficou mais completo em relação ao que foi elaborado inicialmente e as respectivas justificativas quanto à melhora ou piora. Nos próximos tópicos as respostas obtidas são apresentadas na íntegra sem qualquer correção na escrita.

- 1) “Ficou mais claro de ser visualizado cada requisito e sua devida regra de negócio, por poder aplicar dependência”;
- 2) “O uso do apoio computacional no desenvolvimento do Documento de Requisitos permitiu uma abordagem sistêmica na descrição do projeto, frisando a relação entre os diversos requisitos e suas regras de negócios. Por exemplo, quando o software reconhece que a criação de um requisito gera a necessidade da criação de outro, obtém-se um entendimento da coordenação envolvida no sistema, uma visão de vários ângulos”;
- 3) “Sim, os requisitos foram melhores dispostos e apresentados, bem como melhor organizados, facilitando a visualização”;
- 4) “Com o apoio computacional tivemos a possibilidade de identificar alguns erros no documento de requisitos e corrigi-los. Identificamos alguns elementos que faltavam e os adicionamos ao projeto”;
- 5) “Ficou mais claro e mais objetivo”;
- 6) “Foi mais difícil escrever o que nós queríamos, além de que ficou menos organizado. Porém só foi fácil usar o apoio computacional depois que vimos o tutorial”;
- 7) “Sim, o documento feito utilizando o apoio computacional, nos possibilitou detalhar melhor os nossos requisitos funcionais, fornecendo um padrão de descrição mais completo em comparação ao que foi feito no primeiro documento”;
- 8) “Acabou ficando mais completo, pois o apoio computacional exige que os requisitos sejam apresentados de forma muito mais detalhada”.

Com as respostas obtidas referentes à completude do DR, constatou-se que a aplicação do apoio computacional deixou o DR mais completo. Os fatores evidenciados pelos estudantes foram: a) a possibilidade de definir relações de dependência entre os requisitos e regras de negócio; b) a criação de um requisito sugere a criação de outro; c) os requisitos ficaram mais organizados; d) os padrões proporcionaram a identificação de requisitos que não haviam sido identificados no DR inicial; e e) os padrões forneceram uma descrição mais completa para os requisitos.

Com relação à usabilidade, foram feitas várias sugestões para a melhoria do apoio computacional, como: a) aumentar a quantidade de caracteres no campo visão geral do projeto; b) atualizar os parâmetros com os valores fornecidos pelo engenheiro de software durante o seu preenchimento, na primeira versão a atualização ocorria apenas ao salvar o requisito; c) apresentar as sugestões de valores de parâmetros em ordem alfabética; d) aprimorar as dimensões dos quadros e janelas; e) possibilitar a visualização de um requisito do projeto em desenvolvimento por meio da Aba Requisitos, a princípio só era possível a visualização dos requisitos por meio do DR elaborado.

Todas essas sugestões foram implementadas no apoio computacional antes da realização do estudo de caso 2.

Os resultados das questões quantitativas foram os seguintes:

- na média dos grupos, 93% dos requisitos foram escritos com o apoio computacional;
- 87% dos participantes (ou dos grupos) afirmaram que o DR elaborado com os padrões ficou mais completo, com novos requisitos e regras de negócio. Por exemplo, o grupo que desenvolveu o sistema de biblioteca não tinha especificado inicialmente a regra de negócio para estabelecer que o usuário que solicitou a reserva de um item tem prioridade em relação ao usuário que deseja renovar;
- 75% avaliaram que o apoio computacional foi de fácil utilização.

5.3 Estudo de Caso 2

No segundo estudo participaram sessenta estudantes do BCC-UFSCar com conhecimentos mais avançados sobre requisitos e desenvolvimentos de sistemas. Os estudantes foram organizados em 15 grupos com quatro estudantes em cada grupo.

Na disciplina em que esse estudo de caso foi realizado, estava previsto o desenvolvimento de projetos de SI para dispositivos móveis. Apesar do apoio computacional ainda não atender aos requisitos não funcionais, o estudo de caso foi conduzido com a intenção de identificar a abrangência dos padrões de requisitos funcionais para esses sistemas. Os sistemas elaborados foram para: auxiliar pacientes a tomar remédios na data/hora correta, preenchimento de opiniões e questionamentos, compartilhamento e avaliação de livros, consulta de projetos etc. Todos os sistemas requeriam a utilização de pelo menos dois recursos (agenda, galeria de imagens, alarme, etc.) disponíveis no sistema operacional Android. As atividades foram divididas em quatro etapas, sendo:

- 1) Os padrões de requisitos e o apoio computacional foram apresentados aos estudantes, entretanto, sem terem elaborado um DR de modo *ad hoc*, tendo em vista o conhecimento dos estudantes na especificação de requisitos. Para apresentar o uso dos padrões e do apoio computacional foi elaborado um exemplo e esse foi desenvolvido juntamente com os estudantes;
- 2) Em seguida foi solicitada a elaboração do DR de um sistema a ser desenvolvido pelo grupo, com a utilização do ferramental apresentado. Essa atividade foi realizada pelo grupo fora da sala de aula, sendo que as dúvidas poderiam ser esclarecidas via e-mail;
- 3) Após essa atividade, os estudantes responderam a um questionário (Apêndice B), a fim de identificar a porcentagem de requisitos escritos com o uso do apoio computacional e se o apoio computacional é de fácil utilização;
- 4) Essa atividade realizada não teve atribuição de nota a fim de não prejudicar os resultados.

Os estudos de casos 2, 3 e 4 além de possuírem o objetivo de identificar a abrangência dos padrões para a elaboração de DR e a usabilidade do apoio computacional, também objetivou compreender a percepção dos estudantes em

relação: ao reúso na elaboração de DR, ao aumento de produtividade e a eficácia do apoio computacional.

Devido ao conhecimento mais avançado dos membros dos grupos, foi solicitado por meio do questionário, que eles descrevessem sobre a usabilidade do apoio computacional, objetivando a identificação de possíveis melhorias. As respostas obtidas são apresentadas na íntegra sem qualquer correção na escrita, conforme os próximos tópicos:

- 1) “Algumas coisas são muito repetitivas como ter que colocar o mesmo atributo não reusado de outro projeto em várias entidades”;
- 2) “A interface é fácil de ser entendida, intuitiva”;
- 3) “É um programa simples, e com opções intuitivas”;
- 4) “Alguns comandos são bem não-intuitivos”;
- 5) “Utilizar o apoio computacional sem a ferramentas de criação de modelos é menos eficiente do que usar um editor de texto somado ao copiar e colar”;
- 6) “O programa possui uma interface simples, porém precisa de algumas melhorias com relação aos nomes e menus”;
- 7) “Algumas partes da interface eram difíceis de serem compreendidas, além de que o trabalho tinha que ser feito quase totalmente de forma mecânica, no sentido de ter que ficar apertando os mesmos botões várias vezes para executar as mesmas tarefas. Mas no geral, o software é realmente excelente”;
- 8) “Ele é fácil de ser usado só que nem tanto intuitivo”;
- 9) “É uma ferramenta intuitiva, fácil de usar e bastante prático”;
- 10) “Alguns elementos de interface mais intuitivos auxiliariam na hora de criar os requisitos. Atualmente a interface é um pouco confusa”;
- 11) “Não possui um fluxo de construção muito bem definido”.

As respostas contribuíram na identificação das seguintes melhorias no apoio computacional: a) a possibilidade de reutilização de detalhes específicos de um requisito, na instanciação de padrões relacionados. Por exemplo: os valores usados no preenchimento de um requisito instanciado por meio do padrão incluir informação, são sugeridos ao instanciar um padrão relacionado, facilitando o preenchimento; b) um dos grupos declarou a importância de criação de modelos (novos padrões) para outros tipos de requisitos. Com relação a esse item é importante ressaltar que nos estudos de caso os estudantes tiveram acesso apenas ao módulo de instanciação dos

padrões de requisitos para a elaboração do DR, não tendo acesso ao módulo de gestão e especificação de padrões. Essa solicitação foi em decorrência da necessidade de elaborar padrões de requisitos não funcionais que são muito comuns em sistemas para dispositivos móveis e ainda não estão disponíveis no apoio computacional; c) foi detectado que uma das operações não estava executando a função esperada com um clique do mouse, eram necessários dois cliques; e d) necessidade de deixar a interface mais intuitiva.

Como os estudos de casos 2 e 3 foram realizados concomitantes, as melhorias identificadas foram implementadas somente após a conclusão dos mesmos.

Os resultados obtidos das questões quantitativas foram os seguintes:

- 46% dos grupos notaram a possibilidade de reusar totalmente os padrões elaborados e 26% acreditam que o reúso é parcial;
- o reúso dos padrões proporcionou na média, em 20% dos grupos a escrita de mais de 90% dos requisitos e 27% especificaram de 50% a 90%;
- 80% concordam que há facilidade no uso do apoio computacional;
- em relação à produtividade, 33% dos grupos consideraram que o apoio computacional melhora a produtividade entre 21% a 50%, 33% consideraram que melhora entre 51% a 89% e 13% mais de 90%.

O questionário utilizado nesse estudo de caso e nos estudos de caso discutidos na seção 5.4 e na seção 5.5 encontra-se disponível no Apêndice B.

5.4 Estudo de Caso 3

No terceiro estudo de caso participaram vinte e quatro estudantes do curso de EE-UFSCar, com conhecimentos iniciais sobre requisitos, sendo organizados em 7 grupos. Os grupos desenvolveram sistemas no domínio de SI, como pedidos, controle de estacionamento, locação de carros, gerenciamento de tráfego aéreo etc. As atividades foram divididas em quatro etapas, sendo:

- 1) Os padrões de requisitos e o apoio computacional foram apresentados aos estudantes, entretanto, sem terem elaborado um DR de modo *ad hoc*. Para apresentar o uso dos padrões e do apoio computacional foi elaborado um exemplo e esse foi desenvolvido juntamente com os estudantes;

- 2) Em seguida foi solicitada a elaboração do DR de um sistema a ser desenvolvido pelo grupo, com a utilização do ferramental apresentado. Essa atividade foi realizada pelo grupo fora da sala de aula, sendo que as dúvidas poderiam ser esclarecidas via e-mail;
- 3) Após essa atividade, os estudantes responderam a um questionário (Apêndice B), a fim de identificar a porcentagem de requisitos escritos com o uso do apoio computacional e se o apoio computacional é de fácil utilização;
- 4) Essa atividade realizada não teve atribuição de nota a fim de não prejudicar os resultados.

Por meio do questionário foi solicitado aos estudantes relatarem sugestões para a melhoria do apoio computacional. Como resultado foram obtidas as seguintes respostas, que são apresentadas sem qualquer correção na escrita:

- 1) “Maior facilidade de uso”;
- 2) “Incluir uma maior variedade de casos que podem ser usados”;
- 3) “Às vezes sua interface pode ser um pouco confusa, sendo necessário um certo tempo para a devida adaptação à utilização do programa. O programa poderia ter uma interface mais amigável para que o primeiro contato do usuário seja menos confuso”;
- 4) “Interface de uso mais amigável para o primeiro contato do usuário”.

Nesse sentido, contactou-se em resumo, duas solicitações: a) a necessidade de melhorar a facilidade de utilização. Essa necessidade também foi evidenciada no estudo de caso 2. As correções evidenciadas foram corrigidas após a conclusão dos estudos de caso 2 e 3; e b) aumentar as possibilidades de reutilização de requisitos a partir de projetos concluídos. Essa solicitação somente é resolvida com a especificação de mais projetos com o apoio computacional. Para a elaboração dos DRs dos estudos de caso, apenas um projeto de Hotel estava armazenado no repositório.

Os resultados obtidos com as questões quantitativas foram os seguintes:

- 70% dos grupos notaram a possibilidade de reusar totalmente os padrões elaborados e 10% acreditam que o reúso é parcial;
- o reúso dos padrões proporcionou na média, em 20% dos grupos a escrita de mais de 90% dos requisitos e 30% especificaram de 50% a 90%;
- 60% concordam que há facilidade no uso do apoio computacional;

- em relação à produtividade, 30% dos grupos consideraram que o apoio computacional melhora a produtividade entre 21% a 50%, 10% consideram que melhora entre 51% a 89% e 30% mais de 90%.

5.5 Estudo de Caso 4

No quarto estudo de caso participaram trinta e quatro estudantes do curso de BSI-UEMG, divididos em 13 grupos com conhecimentos intermediários sobre requisitos e desenvolvimentos de sistemas. Os projetos foram relacionados a sistemas reais desenvolvidos em empresas que os estudantes atuam, como o para registro de boletim de ocorrência, gestão de escola de informática, registro de ordem de serviço, locação de ferramentas, controle de estoque, etc. As atividades foram divididas em quatro etapas, sendo:

- 1) Primeiramente, os estudantes elaboraram um documento de requisitos de modo *ad hoc*, envolvendo poucos requisitos;
- 2) Os padrões de requisitos e o apoio computacional foram apresentados depois que o DR descrito em 1) já estava elaborado. Para apresentar o uso dos padrões e do apoio computacional foi elaborado um exemplo e esse foi desenvolvido juntamente com os alunos;
- 3) Em seguida foi solicitada a elaboração do DR do sistema em desenvolvimento pelo grupo, agora utilizando o ferramental apresentado. Essa atividade foi realizada pelo grupo fora da sala de aula, sendo que as dúvidas poderiam ser esclarecidas via e-mail;
- 4) Após essa atividade, os estudantes responderam a um questionário (Apêndice B), a fim de identificar a porcentagem de requisitos escritos com o uso do apoio computacional, se o DR ficou mais completo, se o apoio computacional é de fácil utilização e se houve melhora na produtividade;
- 5) Essa atividade realizada não teve atribuição de nota a fim de não prejudicar os resultados.

Foi solicitado aos grupos por meio do questionário a opinião quanto a produtividade do apoio computacional. Nos próximos tópicos as respostas obtidas dos grupos são apresentadas na íntegra sem qualquer correção na escrita.

- 1) “Diminui o tempo gasto na elaboração do documento”;
- 2) “Não precisa re-escrever partes, o software faz o esqueleto para você”;
- 3) “Além de poder reutilizar requisitos já definidos, ele torna a tarefa muito mais dinâmica e simples e possibilita também a edição e correção sem grandes dificuldades”;
- 4) “O software agiliza e facilita o entendimento”;
- 5) “Se perde um pouco de tempo ajustando a necessidade do projeto, mas ajuda bastante”;
- 6) “Ele ajuda no entendimento do documento de requisitos seguindo um padrão simples e bem explicado”;
- 7) “Melhora bem a produtividade mas possui alguns elementos que não são possíveis de se abstrair no apoio computacional”.
- 8) “Às vezes é confuso o uso”.

Com os resultados obtidos nessa questão, foi possível constatar os seguintes fatos: a) há redução no tempo gasto na elaboração do DR; b) fornece solução para um problema, com a necessidade de especificar apenas os detalhes específicos; c) a possibilidade de reutilizar requisitos já definidos torna a tarefa mais simples e dinâmica; d) proporciona agilidade e melhor entendimento do DR; e e) um dos grupos declarou a melhora de produtividade, no entanto, percebeu a necessidade de padrões de requisitos não funcionais.

Os resultados obtidos das questões quantitativas foram:

- 54% dos grupos notaram a possibilidade de reusar totalmente os padrões elaborados e 38% acreditam que o reúso é parcial;
- o reúso dos padrões proporcionou na média, em 38% dos grupos a escrita de mais de 90% dos requisitos; 46% especificaram de 50% a 90%;
- 61% concordam que há facilidade no uso do apoio computacional;
- em relação à produtividade, 8% dos grupos consideraram que o apoio computacional melhora a produtividade entre 21% a 50%, 46% consideram que melhora entre 51% a 89% e 38% mais de 90%.

5.6 Considerações Finais

No primeiro estudo de caso, os estudantes inicialmente escreveram os requisitos de forma *ad hoc* e somente depois reescreveram com o apoio computacional. Os resultados demonstraram que o DR ficou mais completo e que 93% dos requisitos foram possíveis de serem reescritos com o apoio computacional. Entretanto, após a análise dos DRs, identificou-se que os requisitos que não foram escritos com o uso dos padrões foram devido à falta de conhecimento mais aprofundado dos padrões. Os participantes responderam que a especificação dos requisitos com o uso do apoio computacional resultou em um documento mais completo, uma vez que sugestões de requisitos foram fornecidas, os padrões forneceram uma descrição mais completa para a escrita dos requisitos e foi estabelecida a relação de dependência entre os requisitos relacionados.

No segundo estudo de caso, os estudantes de BCC relataram que já haviam especificado DR anteriormente. Os SI elaborados foram para dispositivos móveis. Esse tipo de sistema requer a escrita de requisitos não funcionais, que no momento, estão fora do escopo deste trabalho. Esse fator dificultou os participantes na escrita de todos os requisitos. Entretanto, os seguintes resultados foram obtidos: 47% dos participantes conseguiram reusar entre 51% e 100% dos padrões de requisitos e os demais reusaram entre 10% e 50%. Apesar do resultado não muito satisfatório de reuso dos padrões em decorrência do tipo de SI, 46% dos participantes consideraram que os padrões podem contribuir com o aumento da produtividade entre 51% e 100%. Os estudantes fizeram algumas sugestões para melhorar a usabilidade das funções implementadas no apoio computacional.

No terceiro estudo de caso, os estudantes da engenharia elétrica relataram que não possuíam experiência na elaboração de DR e pouco conhecimento no domínio de SI. Esses fatos resultaram na dificuldade de selecionar o padrão adequado para a escrita dos requisitos. Entretanto, os resultados demonstraram que a utilização do apoio computacional contribuiu para um aumento considerável na produtividade. Além disso, os participantes sugeriram a melhoria da usabilidade do apoio computacional.

As melhorias identificadas no estudo de caso 2 (Seção 5.3) e no estudo de caso 3 (Seção 5.4) foram implementadas após a sua conclusão, tendo em vista que os mesmos ocorreram concomitantes.

No estudo de caso 4, os estudantes do BSI elaboraram um DR de forma *ad hoc* para depois reescreve-lo com o apoio computacional. Nesse estudo, os estudantes relataram a dificuldade de escrever os requisitos de forma *ad hoc*, e que puderam perceber, na prática, o quanto é complexo especificar os requisitos de um sistema completo e atenda às reais necessidades dos *stakeholders*. Esse fato pode ser observado na questão sobre a concordância na reutilização de requisitos, em que 54% dos participantes notaram a possibilidade de reusar totalmente os padrões elaborados e 38% acreditam que o reúso é parcial. Além disso, o resultado sobre reutilização dos padrões para a escrita dos requisitos foi satisfatório: 38% dos participantes especificaram mais de 90% dos requisitos e 46%, de 50% a 90%.

Em relação à produtividade, os participantes do estudo de caso 4 relataram que há redução no tempo gasto na elaboração do DR, pois os padrões fornecem soluções para que engenheiro de software escreva um requisito não a partir do zero, com sugestões para os detalhes específicos da solução e a possibilidade de reutilizar requisitos de outros projetos. Com relação à usabilidade, 61% consideraram que o apoio computacional é de fácil utilização e os problemas evidenciados nos estudos de caso 2 e 3 não mais foram relatados; e 23% não opinaram sobre o assunto. Em conversa com esses estudantes a justificativa para a resposta é que não possuíam o embasamento suficiente para avaliar a questão. Assim, entende-se que os problemas envolvendo essa questão foram resolvidos com as correções realizadas nos outros estudos de caso.

Pelos resultados obtidos após a realização dos estudos de caso e com as respostas obtidas nos questionários há indícios de que os padrões elaborados auxiliaram na construção de um DR mais completo e que há um aumento na produtividade. Esses resultados também contribuíram para a melhoria das funções e da usabilidade do apoio computacional, bem como da completude e da abrangência dos padrões elaborados.

Capítulo 6

CONSIDERAÇÕES FINAIS

6.1 Resultados Obtidos

Este trabalho apresentou uma abordagem para a elaboração de DR no domínio de SI com base em padrões de requisitos, que utiliza reuso de conhecimento de projetos do domínio de SI. Foram elaborados padrões que auxiliam principalmente aos engenheiros de software menos experientes, fornecendo uma base para a elaboração de DR mais completos e consistentes.

Os padrões elaborados atendem às principais características existentes em SI como transações, regras de negócio e a geração de relatórios. Porém, ressalta-se que um conjunto de padrões para manutenção de dados em repositórios de dados podem ser utilizados por qualquer tipo de sistema e não somente os de SI.

A motivação para realização deste trabalho foi em razão do contato com profissionais de TI, durante 11 anos de docência. Notou-se que apesar da teoria ser ensinada completamente, quando os estudantes tornavam-se profissionais retornavam com questões básicas para a obtenção de requisitos e com queixas frequentes de que seus clientes não sabiam especificar e alteravam constantemente o que queriam.

Os padrões elaborados permitem que o desenvolvedor de software:

- Não omita detalhes quando o desenvolvimento é feito, pois há sugestão de padrões relacionados ao que está em uso;
- Consultar os *stakeholders*, com perguntas objetivas e ordenadas, de forma a obter detalhes dos requisitos desejados;

- Definir regras de negócios rotineiras em um SI, bem como outras podem ser criadas em forma de padrões e adicionadas ao catálogo criado.

Considerando que nem sempre a teoria apresentada é usada completamente, torna-se necessário, sempre que possível, criar um apoio computacional para auxiliar as tarefas do desenvolvedor de software. Assim, neste projeto, para facilitar o uso dos padrões de requisitos foi desenvolvido um apoio computacional em uma plataforma Java com o banco de dados MySQL que direciona o engenheiro de software no uso dos padrões para a elaboração de um DR completo e consistente. O apoio computacional proporciona ao engenheiro de software os seguintes recursos:

- Um catálogo de padrões para a identificação e para a escrita completa e consistente dos requisitos obtidos dos *stakeholders*, gerando um DR padronizado. Os padrões de requisitos atendem às funções relacionadas à operacionalização e gestão de negócios, como: a manutenção de dados, o processamento de transações, a emissão de relatórios gerenciais e o estabelecimento de condições de funcionamento do sistema.
- A possibilidade de reutilização de requisitos de projetos concluídos no desenvolvimento de novos projetos. Esses requisitos pertencem aos projetos que foram especificados por meio de padrões de requisitos e armazenados em um repositório;
- O fornecimento de sugestões de preenchimento para os parâmetros do *template* da solução, fornecendo suporte ao engenheiro de software no preenchimento de detalhes específicos do negócio. Essas sugestões foram coletadas de DR obtidos para a elaboração deste trabalho;
- O fornecimento de sugestões de padrões de requisitos que complementam a especificação do sistema. Quando o engenheiro de software utiliza um padrão, outros padrões podem ser sugeridos por meio dos padrões relacionados para a completa e consistente elaboração do DR;
- A definição de dependência entre o requisito relacionado instanciado e o requisito de origem é realizada automaticamente e apresentada no DR, ajudando o engenheiro de software na definição da rastreabilidade entre os requisitos;
- A possibilidade do engenheiro de software estabelecer relação de dependência entre quaisquer requisitos escritos;

- A possibilidade de escrita de novos padrões de requisitos para outros domínios e tipos de requisitos a partir da estrutura elaborada, como por exemplo para os requisitos não funcionais que ainda não estão disponíveis. A metodologia utilizada para elaboração dos padrões é apresentada na seção 3.2;

Um processo de utilização também foi estabelecido para fornecer ao engenheiro um roteiro a ser seguido durante a elicitação de requisitos com os padrões e apoio computacional.

Com a realização de estudos de caso com estudantes de cursos de graduação em duas instituições públicas, é possível destacar os seguintes benefícios na utilização do apoio computacional:

- Os padrões elaborados colaboraram para a completude do DR, uma vez que houve o fornecimento de sugestões de padrões relacionados que complementaram a especificação do sistema;
- A facilidade de uso do apoio computacional foi constatada. Também foram evidenciadas melhorias para aumentar a usabilidade, que foram implementadas após a realização dos estudos de caso;
- Proporcionou por meio dos padrões de requisitos uma descrição mais completa para os requisitos escritos.
- Os requisitos ficaram mais organizados nos DR elaborados;
- Proporcionou a identificação de requisitos que não haviam sido identificados no DR inicial;
- Possibilitou a reutilização de requisitos escritos em outros projetos;
- Houve redução no tempo gasto na elaboração do DR;
- O DR foi melhor entendimento pelos participantes;
- Possibilitou a definição de relação de dependência entre os requisitos e regras de negócio.

Nesse sentido, existem indícios que o apoio computacional e os padrões elaborados fornecem ao engenheiro de software todo o subsídio necessário para a completa e consistente elaboração de um DR.

6.2 Limitações da Proposta

O interesse em melhorar a escrita de um DR de forma que fique mais completo e consistente é constante e contínuo, pois sempre há pontos que podem ser incluídos e melhorados. Nesse sentido, este trabalho apresenta algumas limitações:

- A ausência de padrões para a escrita de requisitos não funcionais. Existem trabalhos que tratam desse tipo de requisitos e podem ser incorporados a esse, após estudo e avaliação;
- Pelo fato da não inclusão de requisitos não funcionais para a escrita dos DR, houve dificuldade na avaliação dos estudos de caso, principalmente naquele que tratou da especificação de sistemas para dispositivos móveis, tendo em vista a alta demanda desse tipo de requisito;
- Outro fator limitante foi o tempo insuficiente para a realização de um estudo de caso em empresas de desenvolvimento. Há necessidade de aplicar os padrões em empresas que desenvolvem esse tipo de sistema e que reusam muitos requisitos de outros sistemas por elas desenvolvidos, para avaliar se os percentuais evidenciados nos trabalhos acadêmicos são os mesmos ou próximos dos profissionais.

6.3 Trabalhos Futuros

A continuidade deste projeto é necessária para possibilitar o desenvolvimento de sistemas de porte maior que os desenvolvidos na academia. Como sugestão de trabalhos futuros pode-se citar:

- A elaboração e inserção de novos padrões de requisitos funcionais no apoio computacional a fim de aumentar a taxa de reúso;
- O uso mais intenso do apoio computacional pode ser realizado a fim de identificar possíveis melhorias quanto à funcionalidade e usabilidade;
- Estudos mais aprofundados podem ser realizados a fim de que possam ser identificados e reusados padrões de requisitos funcionais não contemplados neste trabalho e especificados por outros autores;

- A elaboração de padrões de requisitos não funcionais, bem como a realização de experimentos controlados a fim de verificar a eficácia da proposta;
- A implementação no apoio computacional de outras seções para o DR, com a possibilidade de descrever outras informações sobre o sistema, usando como referência o *template* fornecido pela norma IEEE Std 830;
- A implementação no apoio computacional da gestão de modificações de DRs;
- A aplicação em empresas reais para avaliar taxas de reuso, usabilidade do apoio computacional e dos padrões propostos, facilidade de acrescentar outros padrões a partir das necessidades e costumes existentes nas empresas;
- A implementação no apoio computacional de uma função para medir a taxa de reuso dos padrões por projeto;
- A implementação no apoio computacional de uma função para realizar estimativas de custo e tempo para o desenvolvimento dos projetos especificados, usando ponto de função;
- A integração do apoio computacional a outros ambientes existentes que tenham como uma de suas atividades a elaboração de DRs.

REFERÊNCIAS

- BOEHM, B.; BASILI, V. R. Software Defect Reduction Top 10 List. **IEEE Computer Society**, 2001.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML - Guia do Usuário**. Rio de Janeiro: Campus, 2006.
- BRAGA, R. T. V.; GERMANO, F. S. R.; MASIERO, P. C. GRN: Uma Linguagem de Padrões para Gestão de Recursos de Negócios. **Pattern Languages of Programs Conference**, 1999.
- CHERNAK, Y. Requirements Reuse: The State Of The Practice. **IEEE International Conference on Software Science, Technology and Engineering**, 2012.
- CHRISTOPHER, A. **Uma Linguagem de Padrões**. Porto Alegre: Bookman, 2013.
- FATWANTO, A. Software Requirements Specification Analysis Using Natural Language Processing Technique. **International Conference on Quality in Research**, 2013. 105-110.
- FERREIRA, D. A.; SILVA, A. R. An Enhanced Wiki for Requirements Engineering. **Euromicro Conference on Software Engineering and Advanced Applications**, 2009.
- FRANCH, X. et al. Constructing and Using Software Requirement Patterns. In: _____ **Managing Requirements Knowledge**. : Springer, 2013. p. 95-116.
- GAMMA, E. et al. **Padrões de Projeto**. Porto Alegre/RS: Artmed, 1995.
- IEEE. **Standard Glossary of Software Engineering Terminology**. IEEE Computer Society. New York. 1990.
- IEEE. **Recommended Practice for Software Requirements Specifications**. IEEE Computer Society. New York. 1998.
- JAVA. Disponível em: <http://www.java.com/pt_BR/>. Acesso em: 2 Janeiro 2015.
- KETABCHI, S.; SANI, N. K.; LIU, K. A Norm-Based Approach towards Requirements Patterns. **IEEE Annual Computer Software and Applications Conference**, 2011.
- LAUDON, K. C.; LAUDON, J. P. **Sistemas de Informação Gerenciais**. São Paulo: Pearson Prentice Hall, 2007.
- LI, Y. et al. Requirements patterns for seismology software applications. **IEEE International Workshop on Requirements Patterns**, 2012.

MYSQL. MySQL Documentation: MySQL Reference Manuals. Disponível em: <<http://dev.mysql.com/doc/>>. Acesso em: 02 Janeiro 2015.

NIAZI, M.; SHASTRY, S. Role of requirements engineering in software development process: an empirical study. **Multi Topic Conference, 2003. INMIC 2003. 7th International**, 2003.

PÁDUA, W. **Engenharia de Software: Fundamentos, Métodos e Padrões**. Rio de Janeiro: LTC, 2003.

PALOMARES, C. et al. A catalogue of non-technical Requirement Patterns. **IEEE International Workshop on Requirements Patterns**, 2012.

PALOMARES, C. et al. A Catalogue of Functional Software Requirement Patterns for the Domain of Content Management Systems. **ACM Symposium on Applied Computing**, 2013.

PALOMARES, C.; FRANCH, X.; QUER, C. Requirements Reuse and Patterns: A Survey. In: _____ **Requirements Engineering: Foundation for Software Quality**. Switzerland: Springer, 2014.

PALOMARES, C.; QUER, C.; FRANCH, X. PABRE-Man: Management of a requirement patterns catalogue. **IEEE Software Engineering Conference**, 2011.

PRESSMAN, R. S. **Engenharia de Software - Uma Abordagem Profissional**. 7ª. ed. Porto Alegre/RS: McGraw-Hill, 2011.

REZENDE, D. A. **Engenharia de Software e Sistemas de Informação**. Rio de Janeiro: Brasport, 2005.

ROHER, K.; RICHARDSON, D. Sustainability requirement patterns. **IEEE International Workshop on Requirements Patterns**, 2013.

SHALLOWAY, A.; TROTT, R. J. **Explicando Padrões de Projeto**. Porto Alegre/RS: Bookman, 2004.

SOMMERVILLE, I. **Engenharia de Software**. São Paulo: Pearson Addison-Wesley, 2007.

STAIR, R. M.; REYNOLDS, G. W. **Princípios de Sistemas de Informação**. São Paulo: Pioneira Thomson Learning, 2006.

WAHONO, R. S.; CHENG, J. Extensible requirements patterns of web application for efficient web application development. **IEEE International Symposium on CyberWorlds**, 2002.

WEI, C.; XIAOHONG, B.; XUEFEI, L. A Study on Airborne Software Safety Requirements Patterns. **IEEE International Conference on Software Security and Reliability Companion**, 2013.

WIEGERS, K.; BEATTY, J. **Software Requirements**. Redmond, Washington: Microsoft Press, 2013.

WITHALL, S. **Software Requirement Patterns**. Redmond, Washington : Microsoft Press, 2007.

APÊNDICE A

QUESTIONÁRIO DO ESTUDO DE CASO 1

Questionário de Avaliação dos Padrões para o Domínio de Sistemas e Informação e do Apoio Computacional para a Elaboração do Documento de Requisitos

Nome: _____

1. Quantos requisitos funcionais (RF, RN e Relatórios) têm em seu projeto original?
Número de RF: _____
Número de RN: _____
Número de Relatórios: _____
2. Quantos requisitos do seu projeto original puderam ser especificados usando o apoio computacional com o uso dos Padrões de Requisitos de Software?
Especificar o número de RF, RN e Relatórios.
Número de RF: _____
Número de RN: _____
Número de Relatórios: _____
3. Quantos requisitos vocês precisaram criar porque não existiam no apoio computacional? Especificar o número de RF, RN e Relatórios.
Número de RF: _____
Número de RN: _____
Número de Relatórios: _____

Cite o nome dos requisitos criados: _____

4. O documento elaborado com o apoio computacional ficou mais completo do que o que o grupo criou inicialmente? Justificar sua resposta em termos do que melhorou ou piorou.

() Sim () Não

Justificativa: _____

5. Na opinião do grupo o apoio computacional é fácil de ser usado?

() Sim () Não

6. Na opinião do grupo o apoio computacional é fácil de ser entendido?

() Sim () Não

7. Na opinião do grupo o apoio computacional apresentou problemas? Quais?

() Sim () Não

Cite quais: _____

8. Quanto tempo do grupo demorou para elaborar o documento de requisitos com o apoio computacional?

9. Quantos componentes do grupo instalaram e usaram o apoio computacional efetivamente?

10. Se na resposta da Questão 9 for mais do que um, os documentos de requisitos criados foram diferentes?

() Sim () Não

Em que? _____

11. Quais as sugestões de modificações que o grupo aponta para o apoio computacional a fim de que (mesmo que mais do que um componente do grupo tenha instalado e usado o apoio computacional, coloque todas as sugestões aqui):

a) seja mais fácil de ser usado?

b) seja mais fácil de ser entendido?

c) auxilie mais na elaboração do DR quanto aos RF, RN e Relatórios?

APÊNDICE B

QUESTIONÁRIO DOS ESTUDOS DE CASO 2, 3 E 4

Questionário de Avaliação dos Padrões para o Domínio de Sistemas e Informação e do Apoio Computacional para a Elaboração do Documento de Requisitos

Nome: _____

1. Qual é a sua experiência em desenvolvimento de software?

- 1 ano ou menos
- 2 a 4 anos
- 5 a 10 anos
- mais de 10 anos
- acadêmica

2. Qual é atualmente sua atividade profissional?

- Analista de Requisito
- Programador
- Gerente de Projeto
- Tester
- Analista de Qualidade
- Estudante
- Outro: _____

3. Você já havia especificado algum Documento de Requisitos anteriormente?

- Sim
- Não

4. Qual a sua percepção de aplicar reúso na elaboração de documentos de requisitos?
- 1 – Discordo totalmente
 - 2 – Discordo parcialmente
 - 3 – Neutro
 - 4 – Concordo Parcialmente
 - 5 – Concordo totalmente
5. No DR elaborado, você conseguiu reutilizar os padrões existentes no apoio computacional?
- 1 – Nenhum reúso
 - 2 – Pouco reúso, menos de 10%
 - 3 – Sim, reúso parcial até 50%
 - 4 – Sim, reúso de mais de 51% e menos de 90%
 - 5 – Sim, reúso de mais de 90%
6. Você considera que o PRS fácil de ser usado?
- 1 – Discordo totalmente
 - 2 – Discordo parcialmente
 - 3 – Neutro
 - 4 – Concordo Parcialmente
 - 5 – Concordo totalmente

Justifique: _____

7. Você considera que o PRS atende às necessidades e expectativas do DR que foi elaborado.
- 1 – Discordo totalmente
 - 2 – Discordo parcialmente, atende 10% dos requisitos
 - 3 – Atende menos de 50% dos requisitos
 - 4 – Atende mais de 51% e menos que 90%
 - 5 – Atende mais de 90%

8. Qual a sua opinião em relação à produtividade do PRS

- 1 – Não interfere na produtividade
- 2 – Não melhora a produtividade, até piora
- 3 – Melhora a produtividade entre 10% a 20%
- 4 – Melhora a produtividade entre 21% a 50%
- 5 – Melhora a produtividade entre 51% a 89%
- 6 – Melhora a produtividade mais de 90%

Justifique _____

9. O PRS apresentou problemas?

- Sim Não

Quais? _____

10. Qual a sua sugestão para melhorar o PRS?

