

# Especificação de Requisitos no Domínio de Sistemas de Informação com o Uso de Padrões

Leonardo Barcelos<sup>1,2</sup>, Rosângela Penteadó<sup>1</sup>

<sup>1</sup> Departamento de Computação – Universidade Federal de São Carlos, UFSCar – Programa de Pós-Graduação em Ciência da Computação – Campus São Carlos/SP – Brasil

<sup>2</sup> Departamento de Ciências Exatas e da Terra – Universidade do Estado de Minas Gerais, UEMG – Bacharelado em Sistemas de Informação – Unidade Frutal/MG – Brasil  
{leonardo.barcelos, rosangela}@dc.ufscar.br

**Abstract.** The requirement elicitation is the first step to obtain quality software, requiring constant interaction among stakeholders. The requirements in the information systems domain have similar characteristics, such as the CRUD operations. Reusing knowledge obtained in previous projects, requirement patterns can be elaborated to facilitate the complete and consistent specification of system requirements in that domain. This paper shows a set of functional requirement patterns and business rules elaborated to help the software engineer in the writing of the requirements document. The non-functional requirements were not considered in this project. A computational support was developed, based on that set of patterns, to facilitate the organized and complete writing of the requirements document. Case studies are presented with the use of the computational support.

**Keywords:** requirements document; requirement patterns; system information domain; computational support based on requirement patterns

## 1 Introdução

**Contextualização:** Problemas relacionados com a Engenharia de Requisitos (ER) são amplamente reconhecidos por afetar a qualidade do software e impactar a eficácia do processo de desenvolvimento de software [1]. Estima-se que descobrir e corrigir um problema após a entrega do software pode ser 100 vezes mais caro do que se essa correção ocorrer durante as fases iniciais do desenvolvimento [2]. Nesse sentido, há relatos que a completa compreensão e especificação de requisitos estão entre as tarefas mais difíceis enfrentadas por um engenheiro de software [3].

Chernak [4] relata que na indústria de TI, existem três formas principais de documentação de requisitos, sendo a ER tradicional (48%), a dirigida por casos de uso (28%), a de métodos ágeis (11%) e as demais abordagens que representam 13%. Observa-se que a ER tradicional corresponde a 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 [5].

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 do domínio do cliente, em entendê-las completamente [6]. 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.

**Motivação:** A reutilização de software pode contribuir para o aumento da produtividade, uma vez que conduz à: a) melhoria do tempo de colocação do produto no mercado; b) melhoria da qualidade do produto; e c) redução do custo de desenvolvimento [4]. A aplicação de abordagens de reúso desde a fase inicial do projeto pode ajudar aos engenheiros de software na elicitação, validação e documentação dos requisitos do software, tendo como consequência uma especificação completa, consistente e sem ambiguidade [7] [8].

Padrões de software têm sido usados nas mais variadas áreas, para captar e reutilizar o conhecimento previamente adquirido. Seus usos mais conhecidos são na arquitetura, no projeto de componentes e na implementação. Neste trabalho o enfoque é o uso de padrões na engenharia de requisitos.

Um Padrão de Requisito de Software (PRS) corresponde a um artefato que fornece orientações sobre como especificar requisitos, funcionais, não funcionais e regras de negócio, de forma que possam ser reusados em contextos e problemas bem definidos [9].

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 [9]. Outras fontes para identificação de padrões incluem conhecimentos adquiridos a partir de pesquisas de campo, conhecimento individual de engenheiros de software, normas e melhores práticas ou quaisquer outros artefatos de software [10].

Um padrão possui uma estrutura de apresentação que corresponde à formalização e documentação para captar e reutilizar o conhecimento. 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) [11].

O uso de padrões auxilia engenheiros de software na reutilização de soluções de sucesso para desenvolver novos projetos [11], eliminando a redundância de definição de um conjunto de requisitos para cada projeto de software, o que reduz a carga de trabalho e a habilidade necessária para a comunicação entre os *stakeholders*.

**Problema:** 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 especificação do documento de requisitos (DR), para que o mesmo seja completo, consistente e sem ambiguidade. A reutilização de requisitos com o uso de padrões pode ser uma alternativa viável para aumentar a qualidade da especificação dos requisitos no desenvolvimento de software.

Quando o engenheiro de software conta com um apoio computacional para a elicitação de requisitos e a escrita do DR com o uso de padrões, pode-se garantir também a uniformidade das informações, além do aumento da produtividade uma vez que há um guia a ser seguido.

Este trabalho tem por objetivo apresentar um conjunto de padrões elaborados para a especificação de requisitos funcionais e regras de negócio no domínio de Sistemas de Informação (SI), bem como um apoio computacional elaborado com base nesses padrões para a confecção de um DR. Padrões para requisitos não funcionais não estão no escopo deste projeto, sendo que alguns podem ser encontrados em [12].

**Contribuições:** As principais contribuições são: a) um catálogo de padrões para a elicitación e a escrita de requisitos funcionais e de regras de negócio no domínio de SI; b) 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, com instâncias específicas do sistema em elaboração; c) um repositório de requisitos elaborado a partir dos padrões descritos em a); d) a possibilidade de escrita de novos padrões a partir de uma estrutura elaborada.

Atualmente, o processamento de informação corresponde a maior área de aplicação de software, o que justifica a seleção do domínio de SI para a elaboração de padrões propostos neste trabalho. A relevância na elaboração desses padrões justifica-se em razão da similaridade encontrada quando da descrição de diferentes sistemas de SI.

**Organização do trabalho:** Este trabalho está dividido em seis seções, além desta introdução. Na Seção 2 são apresentados os trabalhos relacionados com o uso de padrões na especificação de requisitos. Na Seção 3 são feitas considerações sobre conceitos em SI. Na Seção 4 são apresentados e comentados os padrões elaborados para o domínio de SI. Na Seção 5 é apresentado o apoio computacional desenvolvido para a elaboração de documentos de requisitos com base nos padrões exibidos na Seção 4. Na Seção 6 são apresentados os resultados da avaliação do uso do apoio computacional e dos padrões elaborados. Na Seção 7 são comentadas as considerações finais e sugestões de trabalhos futuros.

## 2 Trabalhos Relacionados

Withall [9] apresenta um catálogo com trinta e sete padrões de requisitos organizados em oito domínios (fundamental, informação, entidade de dados, função do usuário, desempenho, flexibilidade, controle de acesso e comercial). Cada padrão atende a uma funcionalidade específica desses domínios, porém, alguns padrões compartilham e integram informações. A maioria dos padrões atende à especificação de requisitos não funcionais e técnicos, sendo alguns também para requisitos funcionais.

Roher e Richardson [10] sugerem a incorporação de padrões de requisitos de sustentabilidade no processo de engenharia de requisitos, a fim de facilitar a escrita do DR para determinados tipos de requisitos de sustentabilidade. Esses padrões, nem sempre são tratados nos SI.

Wei et al. [13] 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. [11] e os padrões de requisitos de Withall [9]. 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. [14] 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. [15] apresentam o *framework* PABRE (*P*attern-*B*ased *R*equirements *E*licitation), 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. Esses padrões atendem principalmente a requisitos não funcionais e de gestão de documentos, que estão fora do escopo deste projeto.

O estudo de Palomares et al. [16] apresenta dois padrões de requisitos não técnicos, elaborados por meio do *framework* PABRE. Além disso, os autores 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. Esses padrões de requisitos funcionais não são aplicáveis ao domínio de SI comerciais.

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 [17]. 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.

### **3 Conceitos em Sistemas de Informação**

Em documentos de requisitos de SI é comum encontrar requisitos funcionais que especificam operações para processamento, manutenção e consulta de dados, tais como criar, ler, atualizar e excluir informação.

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) [18]. Os SPTs são usados para registrar transações básicas da organização, tais como registro de pedidos de venda, de reservas de hotel, de vendas, de manutenção do registro do cliente etc. Os SIGs usam os dados armazenados pelos SPTs para fornecer informações aos gerentes e tomadores de decisões, como exemplo pode-se citar um relatório de pedidos semanal e a lista de reservas do hotel em 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 [19].

Os requisitos funcionais desses tipos de sistemas, na maioria dos casos, requerem a definição de regras de negócio específicas. Na perspectiva de SI, uma regra de negócio é uma declaração que define ou restringe alguma circunstância do negócio [20].

Wieggers e Beatty [20] identificam cinco tipos principais de esquemas de classificação de regras de negócio, que apresentam 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 devolve 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 a classificação acima não foi utilizada. Contudo, o engenheiro de software deve conhecê-los de forma a elaborar um DR completo e sem ambiguidade, observando a especificação dos requisitos funcionais, não funcionais e das regras de negócio de um sistema de software.

Na seção seguinte é apresentada a metodologia usada para a elaboração dos padrões de requisitos.

## 4 Elaboração de Padrões de Requisitos de Software para o Domínio de Sistemas de Informação

A metodologia usada para a elaboração dos padrões constou 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 (inclusão, remoção, alteração, consulta), processamento de transação (compra, venda, 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 não possuíam ambiguidade, bem como identificar as partes do texto que deveriam ser fixas e as partes variáveis.
4. **Especificação do Padrão:** foi definida a aplicabilidade e a solução do padrão para atender a especificação de um conjunto de requisitos similares (Quadro 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).
5. **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.

Para atender ao domínio de SI, neste projeto, foi estabelecida uma estrutura de apresentação de padrões com base na estrutura do Gamma et al. [11], apresentada no Quadro 1.

**Quadro 1.** Estrutura Adotada para Apresentação dos Padrões

Elemento	Descrição
<b>Nome</b>	Especifica o nome do padrão, que deve ser único e refletir a aplicabilidade do padrão.
<b>Domínio</b>	Corresponde ao domínio de aplicação do padrão.
<b>Propósito</b>	Descreve o objetivo da aplicação do padrão.
<b>Problema</b>	Descreve a situação em que o padrão pode ser aplicado.
<b>Consequência</b>	Descreve as consequências de se utilizar o padrão.
<b>Tipo</b>	Especifica o tipo de requisito: funcional, não funcional ou regra de negócio.
<b>Solução</b>	Apresenta um <i>template</i> para a especificação da parte fixa e variável do requisito que o padrão deve representar. A parte fixa apresenta um texto padrão para a especificação 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.
<b>Padrões Relacionados</b>	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.

A seguir são apresentados dois dos padrões elaborados a partir dos requisitos obtidos: Processar Transação (processamento de transação) no Quadro 2 e Condição de Execução (regra de negócio) no Quadro 3.

**Quadro 2.** Padrão de Requisito Funcional: Processar Transação

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

**Quadro 3.** Padrão de Regra de Negócio: Condição de Execução

<b>Nome</b>	Condição de Execução
<b>Domínio</b>	Sistemas de Informação
<b>Propósito</b>	Estabelecer as condições permitidas para uma determinada operação.
<b>Problema</b>	Especificar as condições específicas da operação.
<b>Consequência</b>	A especificação de um requisito que descreve uma regra de negócio que estabelece as condições de execução de uma determinada operação.
<b>Tipo</b>	Regra de Negócio
<b>Solução</b>	<p><b>Template:</b> A &lt;operação&gt; somente deve ser permitida se &lt;condição&gt;</p> <p><b>Parâmetro:</b> &lt;operação&gt; nome da operação que deseja estabelecer uma condição (1). &lt;condição &gt; descrição da condição para a execução da operação (1..*).</p> <p><b>Sugestões de Parâmetros:</b> &lt;operação&gt; reserva &lt;condição&gt; se houver estoque suficiente, houver vagas suficientes para atende-la.</p> <p><b>Exemplos da Descrição do Requisito:</b> - A reserva somente deve ser permitida se houver vagas suficientes para atende-la.</p>
<b>Padrões Relacionados</b>	-

No Quadro 4 é apresentado o catálogo de padrões elaborado para o domínio de SI que atende às operações básicas de manutenção e consulta de dados, o processamento de transações, a consulta/impressão de relatórios gerenciais e o estabelecimento de regras

de negócios. Na primeira coluna consta o tipo do padrão, na segunda tem-se o nome do padrão, e a consequência é exibida na terceira coluna.

Não foram gerados padrões para atender aos requisitos para SADs em razão da especificidade que eles possuem.

**Quadro 4.** Catálogo de Padrões de os para o Domínio de Sistemas de informação

<b>Tipo</b>	<b>Nome</b>	<b>Consequência</b>
<b>Operações Básicas</b>	Incluir Informação	A especificação de um requisito que descreve a necessidade de um usuário de armazenar informação de uma entidade de dados.
	Alterar Informação	A especificação de um requisito que descreve a necessidade de um usuário de alterar determinados dados de uma entidade de dados.
	Excluir Informação	A especificação de um requisito que descreve a necessidade de um usuário de excluir informação de uma entidade de dados.
	Recuperar Informação	A especificação de um requisito que descreve a necessidade de um usuário de recuperar informação de uma entidade de dados por meio da descrição de parâmetros.
<b>Processamento de Transações</b>	Processar Transação	A especificação de um requisito que descreve a necessidade de um usuário de processar uma transação e armazenar sua informação.
<b>Informações Gerenciais</b>	Especificar Consulta Gerencial	A especificação de um requisito que descreve a necessidade de um usuário de emitir consultas de informações gerenciais em tela ou impresso.
<b>Regras de Negócio</b>	Informação Requerida	A especificação de um requisito que descreve uma regra de negócio que estabelece as informações requeridas por uma determinada operação.
	Condição de Execução	A especificação de um requisito que descreve uma regra de negócio que estabelece as condições de execução de uma determinada operação.
	Limite de Execução	A especificação de um requisito que descreve uma regra de negócio que estabelece limites específicos de execução de uma determinada operação.
	Calculo de Valor	A especificação de um requisito que descreve a necessidade de um usuário de descrever um determinado cálculo matemático, requerido em uma operação.

## 5 Apoio Computacional Para a Elaboração de Documentos de Requisitos Com o Uso De Padrões

Palomares et al. [21] afirmam que há dificuldade de utilizar padrões sem um apoio computacional para direcionar o engenheiro de software. 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 apoio.

Neste projeto após criar os padrões de requisitos funcionais e de regras de negócio, considerando as observações de Palomares et al. [21], 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. Ressalta-se novamente que os requisitos não funcionais não fazem parte do escopo deste trabalho.

O processo de uso consiste na utilização do apoio computacional para a instanciação dos padrões a fim de elaborar documentos de requisitos de SI com os objetivos anteriormente citados. Os passos que devem ser seguidos são:

1. **Seleção do Padrão:** de acordo com a necessidade do cliente, o engenheiro de software seleciona, no apoio computacional, o padrão adequado para esse tipo de problema. Essa escolha sempre é confirmada com o cliente. Se o padrão atende aos requisitos solicitados, o engenheiro de software avança para o passo seguinte.
2. **Especificação das partes variáveis:** o engenheiro de software informa ao cliente as variabilidades que existem para a solução do padrão selecionado. O cliente pode escolher entre as opções que satisfaz ao seu sistema, fornecendo as informações pertinentes a partir do conjunto de sugestões cadastradas para o padrão no apoio computacional. No entanto, existe a possibilidade do cliente informar o que deseja e essas serem posteriormente incorporadas como variabilidades.
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.

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 e projeto.

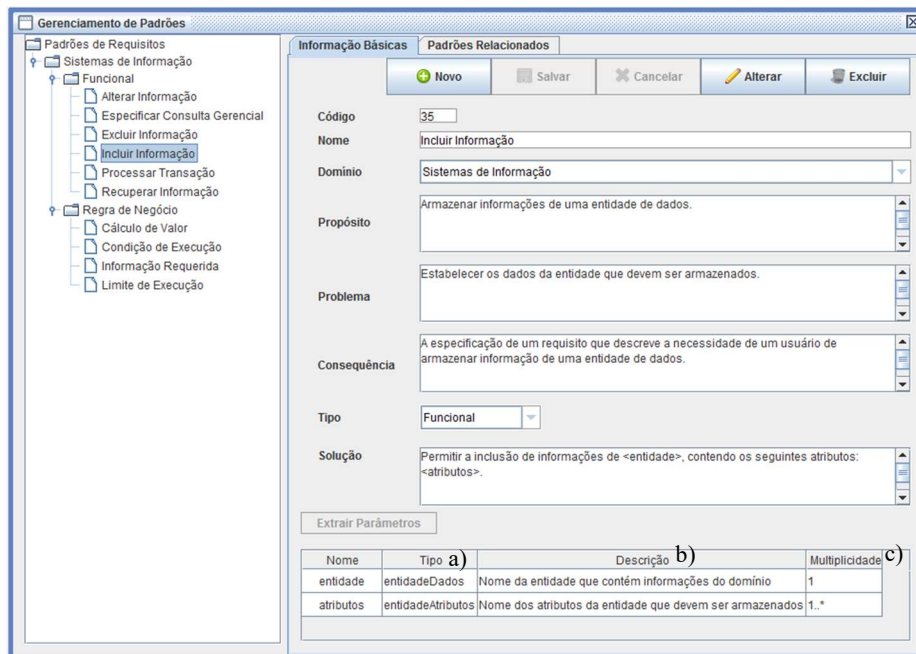
Com o módulo de especificação e gestão de padrões (Fig. 1) é possível realizar a manutenção dos padrões. Na definição do *template* da solução, para cada parâmetro que representa a parte variável do padrão, é fornecido: a) um tipo, que está associado a um conjunto de sugestões de preenchimento que pode 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 realizar a especificação do padrão as abas Informações Básicas e Padrões Relacionados são utilizadas. Os padrões relacionados com o padrão em questão são descritos por meio da aba com o respectivo nome.

O módulo de instanciação de padrões (Fig. 2) possui quatro abas. As mesmas são apresentadas em detalhes nos próximos tópicos:

- **Informações Básicas:** apresenta as informações de aplicabilidade e solução do padrão.



**Fig. 1.** Módulo de Especificação e Gestão de Padrões

- Elaboração:** usada para a especificação de um requisito por meio do padrão escolhido. As letras colocadas na Fig. 2 tem o objetivo de guiar o leitor, assim são as mesmas utilizadas a seguir. Nessa aba é apresentado: 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 Fig 2. 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]. A finalização da especificação de um requisito, ou seja, com o preenchimento de todos os parâmetros, é realizada ao pressionar o botão [Salvar].
- Requisitos:** apresenta todos os requisitos do projeto; permite a seleção do requisito para a edição ou exclusão; permite a geração do DR, no formato *pdf*; permite o estabelecimento de relacionamento (rastreadibilidade) entre os requisitos; e fornece sugestões de padrões relacionados com o requisito instanciado.
- Reúso:** apresenta os requisitos já instanciados em outros projetos, que foram elaborados pelo padrão selecionado. Esses requisitos podem ser reusados no projeto atual ao invés de serem especificados com o uso de um padrão.

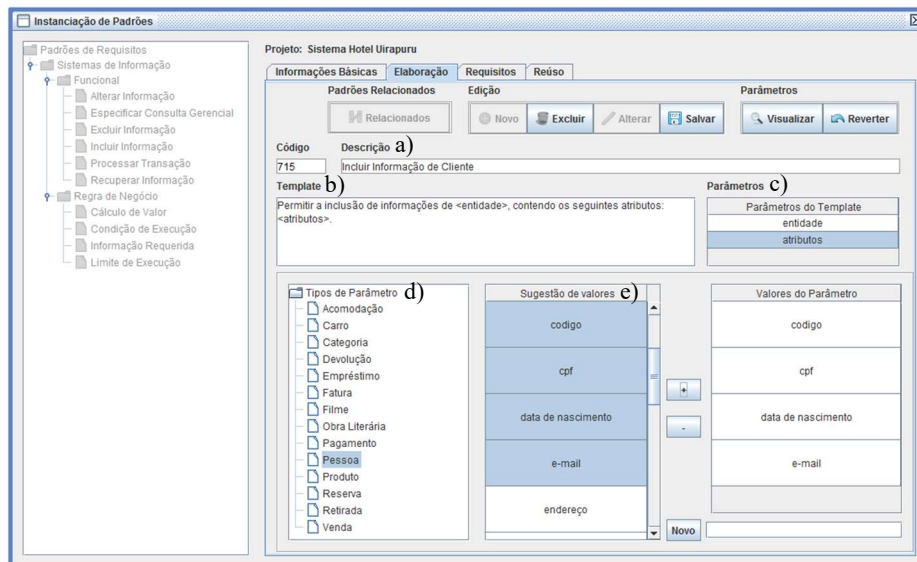


Fig. 2. Módulo de Instanciação de Padrões

Uma versão simplificada do DR resultante da instanciação dos padrões apresentados anteriormente é apresentada na Fig. 3. 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.

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

Fig. 3. Documento de Requisitos de Exemplo

O apoio computacional foi desenvolvido na linguagem Java [22] utilizando o banco de dados MySQL [23].

Este trabalho difere dos demais apresentados na Seção 2, pois: a) permite a escrita padronizada de um DR com requisitos funcionais e regras de negócio 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 de hotel, um sistema de farmácia ou qualquer outro e terá no DR a especificação completa desse requisito; b) com o uso do apoio computacional o DR é elaborado completamente em formato *pdf*; 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 essa função para qualquer 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.

## 6 Avaliação da Aplicação dos Padrões com o Apoio Computacional

Para avaliar a abordagem proposta quanto à completude dos padrões e usabilidade do apoio computacional, foram realizados estudos de casos com estudantes de graduação dos cursos de Bacharelado em Ciência da Computação e de Engenharia Elétrica da UFSCar, campus de São Carlos/SP e do curso de Sistemas de Informação da UEMG, unidade de Frutal/MG.

**Estudo de caso 1:** participaram quarenta e dois estudantes divididos em oito grupos e cada um com um projeto relacionado a serviços oferecidos pela UFSCar como: biblioteca, restaurante universitário, locação de quadra, inscrição em disciplina, etc. Previamente, os estudantes haviam elaborado um DR de modo *ad hoc*, com sistemas simples e com poucos requisitos.

Os padrões de requisitos de software e o apoio computacional foram apresentados, bem como um treinamento com um sistema exemplo com requisitos diferentes dos que constavam dos documentos de requisitos elaborados pelos estudantes.

Em seguida foi solicitada a elaboração do DR do sistema em desenvolvimento pelo grupo, agora utilizando o ferramental apresentado. Após essa atividade, os estudantes responderam a um questionário, a fim de identificar a porcentagem de requisitos especificados com o uso do apoio computacional, se o DR ficou mais completo e se o apoio computacional é de fácil utilização. O resultado obtido foi: na média, 93% dos requisitos foram especificados. Em razão da falta de conhecimento mais aprofundado dos padrões existentes, alguns requisitos não foram especificados por meio deles; 87% dos participantes afirmaram que o DR ficou mais completo, com novos requisitos e regras de negócio; e 75% avaliaram que o apoio computacional foi de fácil utilização.

**Estudo caso 2:** foi realizado em duas disciplinas da UFSCar, sendo uma: i) composta por estudantes da ciência da computação com conhecimentos mais avançados sobre requisitos e desenvolvimento de sistemas para dispositivos móveis e ii) com estudantes do curso de engenharia elétrica, com conhecimentos iniciais sobre requisitos, que desenvolveram sistemas simples. Os estudantes estavam distribuídos em grupos e cada um desenvolvendo sistemas do domínio de SI. Nesse estudo os DR foram elaborados após o conhecimento e treinamento com os padrões de requisitos e apoio computacional, entretanto sem terem escrito o DR de modo *ad hoc*. Os resultados obtidos foram os seguintes:

i) Sistemas desenvolvidos para dispositivos móveis; 15 grupos com quatro estudantes em cada grupo; 46% dos grupos notaram que é possível reutilizar totalmente os requisitos na elaboração do DR e 26% citaram que o reúso é parcial. O reúso dos padrões proporcionou na média, em 20% dos grupos a especificação de mais de 90% dos requisitos; 27% especificaram de 50% a 90%. Além disso, 80% concordam que há facilidade no uso do apoio computacional.

ii) Sistemas de informação simples; 7 grupos com total de 24 estudantes; 70% dos grupos notaram que é possível reutilizar totalmente os requisitos na elaboração do DR e 10% citaram que o reúso é parcial. O reúso dos padrões proporcionou na média, em 20%

dos grupos a especificação de mais de 90% dos requisitos; 30% especificaram de 50% a 90%. Além disso, 60% concordam que há facilidade no uso do apoio computacional.

**Estudo de caso 3:** participaram trinta e quatro estudantes do curso de Sistemas de Informação da UEMG, divididos em treze grupos. Os projetos eram relacionados a sistemas reais desenvolvidos em empresas que os estudantes atuam. Como ocorreu no estudo de caso 1, os DR foram primeiramente elaborados de modo *ad hoc* e após a apresentação dos padrões e do apoio computacional foram refeitos.

Os resultados obtidos foram: 54% dos grupos notaram que é possível reutilizar totalmente os requisitos na elaboração do DR e 38% citaram que o reúso é parcial. O reúso dos padrões proporcionou na média, em 38% dos grupos a especificação de mais de 90% dos requisitos; 46% especificaram de 50% a 90%. Além disso, 61% concordam que há facilidade no uso do apoio computacional.

Há indícios, pelos resultados obtidos após a realização dos estudos de caso e com as respostas obtidas nos questionários, que os padrões elaborados auxiliaram a confecção de um DR mais completo. Outros estudos devem ser realizados, principalmente com sistemas reais em empresas para que resultados mais conclusivos sejam obtidos. Em razão da limitação de espaço o questionário aplicado não é apresentado neste artigo.

## 7 Considerações Finais

Este artigo apresentou uma abordagem para escrita de documentos de requisitos no domínio de SI com base em padrões de requisitos. Essa abordagem foi desenvolvida com base no reúso de conhecimento de projetos do domínio de SI. Os padrões elaborados auxiliam principalmente aos engenheiros de software menos experientes, fornecendo uma base para a escrita de DR mais completos e consistentes.

Para facilitar o uso dos padrões foi criado um apoio computacional em uma plataforma Java com o banco de dados MySQL que direciona o engenheiro de software no uso de padrões e na confecção de um DR com completa checagem da dependência existente entre os requisitos funcionais bem como a completa especificação de regras de negócio. Os requisitos não funcionais ainda não estão disponíveis no apoio apresentado. O apoio computacional, bem como os padrões elaborados e os questionários utilizados nos estudos de caso estão disponíveis no endereço <http://advanse.dc.ufscar.br/tools.html>.

Como citado na Seção 6, os resultados obtidos com a realização de estudos de caso com estudantes de cursos de graduação em duas instituições públicas, evidenciou a facilidade de uso do apoio computacional. Também pode ser notado que os padrões elaborados colaboraram para que os requisitos funcionais fossem especificados com maior segurança, considerando que há a possibilidade de verificação de dependência entre esses requisitos. Dessa forma, um engenheiro de software menos experiente conta com a verificação de dependência entre requisitos e a completa especificação das regras de negócio, ou mesmo do reúso de especificações realizadas em projetos anteriores. Assim, o DR produzido é mais completo e abrangente.

Como trabalhos futuros, novos padrões podem ser elaborados e inseridos no apoio computacional a fim de aumentar a taxa de reúso. Com o uso do apoio computacional mais intenso, podem ser realizadas melhorias quanto a sua funcionalidade e usabilidade. Uma melhoria prevista é a extensão do módulo de instanciação de padrão, permitindo o estudo da incorporação de novos valores para o conjunto de valores sugeridos para os parâmetros, com base nos novos valores fornecidos pelos *stakeholders*. Outra extensão

é a elaboração de padrões que atendam aos requisitos não funcionais. Considera-se também que estudos mais aprofundados sejam realizados a fim de que possam ser reusados requisitos funcionais já especificados por outros autores. A inclusão desses requisitos não funcionais deve ocorrer no apoio computacional bem como a realização de experimentos controlados a fim de verificar a eficácia da proposta.

## Referências

- [1] M. Niazi e S. Shastri, "Role of requirements engineering in software development process: an empirical study," *Multi Topic Conference, 2003. INMIC 2003. 7th International*, 2003.
- [2] B. Boehm e V. R. Basili, "Software Defect Reduction Top 10 List," *Computer (IEEE)*, 2001.
- [3] R. S. Pressman, *Engenharia de Software - Uma Abordagem Profissional*, 7ª ed., Porto Alegre/RS: McGraw-Hill, 2011.
- [4] Y. Chernak, "Requirements Reuse: The State Of The Practice," *IEEE International Conference on Software Science, Technology and Engineering*, 2012.
- [5] A. Fatwanto, "Software Requirements Specification Analysis Using Natural Language Processing Technique," *International Conference on Quality in Research*, pp. 105-110, 2013.
- [6] I. Sommerville, *Engenharia de Software*, São Paulo: Pearson Addison-Wesley, 2007.
- [7] S. Ketabchi, N. K. Sani e K. Liu, "A Norm-Based Approach towards Requirements Patterns," *IEEE Annual Computer Software and Applications Conference*, 2011.
- [8] C. Palomares, C. Quer e X. Franch, "PABRE-Man: Management of a requirement patterns catalogue," *IEEE Software Engineering Conference*, 2011.
- [9] S. Withall, *Software Requirement Patterns*, Microsoft Press, 2007.
- [10] K. Roher e D. Richardson, "Sustainability requirement patterns," *IEEE International Workshop on Requirements Patterns*, 2013.
- [11] E. Gamma, R. Helm, R. Johnson e J. Vlissides, *Padrões de Projeto*, Porto Alegre/RS: Artmed, 1995.
- [12] X. Franch, C. Palomares, C. Quer, S. Renault e F. Lazzer, "A Metamodel for Software Requirement Patterns," em *Requirements Engineering: Foundation for Software Quality*, Springer, 2010.
- [13] C. Wei, B. Xiaohong e L. Xuefei, "A Study on Airborne Software Safety Requirements Patterns," *IEEE International Conference on Software Security and Reliability Companion*, 2013.
- [14] Y. Li, C. Pelties, M. Käser e N. Nararan, "Requirements patterns for seismology software applications," *IEEE International Workshop on Requirements Patterns*, 2012.
- [15] X. Franch, C. Quer, S. Renault, C. Guerlain e C. Palomares, "Constructing and Using Software Requirement Patterns," em *Managing Requirements Knowledge*, Springer, 2013, pp. 95-116.
- [16] C. Palomares, C. Quer, X. Franch, C. Guerlain e S. Renault, "A catalogue of non-technical Requirement Patterns," *IEEE International Workshop on Requirements Patterns*, 2012.
- [17] C. Palomares, C. Quer, X. Franch, S. Renault e C. Guerlain, "A Catalogue of Functional Software Requirement Patterns for the Domain of Content Management Systems," *ACM Symposium on Applied Computing*, 2013a.
- [18] R. M. Stair e G. W. Reynolds, *Princípios de Sistemas de Informação*, São Paulo: Pioneira Thomson Learning, 2006.
- [19] K. C. Laudon e J. P. Laudon, *Sistemas de Informação Gerenciais*, São Paulo: Pearson Prentice Hall, 2007.
- [20] k. Wiegers e J. Beatty, *Software Requirements*, Redmond, Washington: Microsoft Press, 2013.
- [21] C. Palomares, X. Franch e C. Quer, "Requirements Reuse and Patterns: A Survey," em *Requirements Engineering: Foundation for Software Quality*, Springer, 2014.
- [22] Java. [Online]. Available: [http://www.java.com/pt\\_BR/](http://www.java.com/pt_BR/). [Acesso em 2 Janeiro 2015].
- [23] MySQL, "MySQL Documentation: MySQL Reference Manuals," [Online]. Available: <http://dev.mysql.com/doc/>. [Acesso em 02 Janeiro 2015].