Padrões de Requisitos para Sistemas de Registro Eletrônico de Saúde

Mariana Crisostomo Martins[1], Taciana Novo Kudo[1], Renato F. Bulcão-Neto[1]

[1]Instituto de Informática – Universidade Federal de Goiás (UFG)
Caixa Postal 131 – 74.690-900 – Goiânia – GO – Brazil

maricrisostomo.martins@gmail.com, {taciana,rbulcao}@ufg.br

Abstract. *A software requirements standards catalog (CPRS) organizes common requirements between applications in a domain to be reused. Thus, it reduces the efforts of eliciting, specifying, and validating requirements for a new application in the same field. CPRS plays an essential role for Requirements Engineering and the entire life cycle, as more than 50% of defects in projects come from low-quality requirements. Recent studies reinforce that the development of CPRS is one of the actions that promote greater dissemination of the adoption of requirements standards in the industry. However, in addition to the low availability of CPRS, few studies carry out the validation and evaluation of CPRS. Thus, the aim is to develop a CPRS for Electronic Health Record Systems (EHR-S), validated and evaluated by specialists in this application domain. The construction of CPRS is supported by the Terminal Model Editor (TMEd) tool, and the concepts and structure of CPRS follow the definitions of the Software Pattern Metamodel (SoPaMM) metamodel. The construction of the CPRS founds on requirements used in the EHR-S certification by the Brazilian Society of Health Informatics (BSHI). The CPRS will be available to BSHI to support accreditation and the development of EHR-S.*

Resumo. *Um catálogo de padrões de requisitos de software (CPRS) organiza requisitos comuns entre aplicações de um domínio para que sejam reusados e, assim, reduzir os esforços de elicitação, especificação e validação de requisitos de uma nova aplicação de mesmo domínio. O CPRS tem importante papel para a Engenharia de Requisitos e todo o ciclo de vida, pois mais de 50% dos defeitos identificados em projetos advêm de requisitos de baixa qualidade. Estudos recentes reforçam que o desenvolvimento de CPRS é uma das ações que promovem maior disseminação da adoção de padrões de requisitos na indústria. Além da baixa disponibilidade de CPRS, poucos estudos realizam a validação e a avaliação de CPRS. Assim, visa-se desenvolver um CPRS para Sistemas de Registro Eletrônico em Saúde (S-RES), validado e avaliado por especialistas desse domínio de aplicação. A construção do CPRS é apoiada pela ferramenta Terminal Model Editor (TMEd), e os conceitos e a estrutura do CPRS seguem as definições do metamodelo Software Pattern Metamodel (SoPaMM). A construção do CPRS apoia-se em requisitos usados na certificação de S-RES pela Sociedade Brasileira de Informática em Saúde (SBIS). O CPRS será então disponibilizado à SBIS para apoiar não somente a certificação, mas também o desenvolvimento de S-RES.*

Cadernos do IME - Série Informática
e-ISSN: 2317-2193 (online)
DOI: 10.12957/cadinf.2021.64964

1. Problema

Cerca de 56% dos defeitos em um software são advindos de documentações incompletas e baixa qualidade, com requisitos errados, omitidos, conflitantes e ambíguos [Tockey 2015]. Esses contratempos na Engenharia de Requisitos impactam no desenvolvimento do produto com atrasos na entrega e insatisfação dos clientes.

O reúso de requisitos é uma abordagem alternativa ao método tradicional de elicitação para lidar com esses problemas [Chernak 2012, Irshad et al. 2018]. Em uma empresa de desenvolvimento de software, é comum a possibilidade de reusar requisitos recorrentes em diferentes projetos. Dentre as abordagens de reúso existentes [Irshad et al. 2018], padrões de requisitos de software (PRS) têm sido amplamente estudados recentemente [Withall 2007, Palomares et al. 2013, Kudo et al. 2019]. Um PRS é uma abstração que agrega comportamentos e serviços comuns a vários sistemas, e podem ser reutilizados em software similar [Withall 2007].

A elicitação de requisitos com o apoio de PRS permite maior agilidade no processo, pois PRS reduz as preocupações do analista como: tempo, custo e qualidade das especificações com requisitos completos, consistentes e uniformes [Withall 2007, Palomares et al. 2014]. O uso de PRS também possibilita o melhor gerenciamento de requisitos porque permite maior orientação e vários níveis de detalhes oportunizando rastreabilidade e manutenibilidade por meio da estrutura com informações a respeito do desenvolvimento, teste e uso dos padrões. Ao especificar um PRS, segue-se um *template* que organiza e uniformiza a escrita dos requisitos, de modo a torná-los mais completos, consistentes e de maior qualidade [Withall 2007]. Em suma, são vantagens de uso de PRS: aumento da consistência e qualidade das especificações, melhor gerenciamento de requisitos e menor tempo de elicitação [Palomares et al. 2014].

A abordagem de PRS pode ser adequada ao tipo de requisito e/ou domínio de aplicação. O catálogo de padrões de requisitos (CPRS) é uma maneira estruturada de agrupar PRS segundo alguma categoria. Na prática, para o analista, um CPRS facilita o reúso de conhecimento de forma organizada e estruturada.

Um estudo terciário recente [Kudo et al. 2020] apresenta uma agenda de pesquisa sobre PRS, em que aponta que desenvolver e publicizar CPRS é uma das ações necessárias para reduzir a lacuna entre a teoria e a prática de PRS na indústria.

Apesar de sua importância, existem poucos CPRS disponíveis, como aqueles para sistemas em nuvem [Beckers et al. 2014], gerenciamento de conteúdo [Palomares et al. 2012] e licitações [Costal et al. 2019]. Além da *baixa disponibilidade de CPRS*, há *pouca informação na literatura sobre o ciclo de desenvolvimento* de um CPRS e a maioria desses estudos *não descreve como catálogos são validados* [Kopczynska et al. 2018, Beckers et al. 2014, Amorndettawin and Senivongse 2019] e *avaliados* [Riaz et al. 2016, Palomares et al. 2013].

Sistemas de Registro Eletrônico em Saúde (S-RES) precisam ser avaliados e atestados quanto a suas funcionalidades, estrutura, conteúdo, segurança, aderência a legislações, etc. Os requisitos de S-RES no Brasil foram elaborados por especialistas e reunidos em um manual que orienta o desenvolvimento e a certificação de S-RES, segundo a Sociedade Brasileira de Informática em Saúde (SBIS) [Kiatake et al. 2020]. Os requisitos presentes no manual poderiam apresentar uma estrutura, organização e linguagem mais facilitada.

Em suma, o CPRS que tem sido desenvolvido para S-RES pode contribuir tanto na especificação de requisitos quanto na certificação desses sistemas, além das vantagens inerentes de PRS, como a melhora na qualidade das especificações com menor esforço.

2. Background

Esta seção apresenta um metamodelo para representação de padrões de software, uma ferramenta para construção de CPRS e o manual de requisitos de S-RES.

2.1. Software Pattern Metamodel (SoPaMM)

O metamodelo SoPaMM fornece construtores para o desenvolvimento de catálogos de padrões de software em geral [Kudo et al. 2019]. O ponto central do SoPaMM é o modo como representa padrões de requisitos e a sua interligação com padrões de testes. Para isso, são reusados conceitos e práticas da abordagem de desenvolvimento ágil *Behaviour-Driven Development* (BDD) [Chelimsky et al. 2010].

No SoPaMM, padrões de requisitos funcionais são decompostos em *features* associadas a histórias de usuário (AS <interessado>, I CAN <o quê?>, SO THAT <para quê?>). Uma história de usuário pode ter múltiplos cenários descritos na sintaxe *Given* (pré-condições do cenário), *When* (passos de execução do cenário) e *Then* (comportamentos esperados do cenário) da linguagem *Gherkin*.

Padrões de teste de aceitação são interligados a padrões de requisitos funcionais por meio do relacionamento entre casos de teste e cenários de execução. Nesse relacionamento são estabelecidos os dados de teste de entrada e saída que deverão ser utilizados para verificar os comportamentos de cada cenário vinculado a um padrão de requisito funcional.

2.2. Terminal Model Editor (TMEd)

Desenvolvida sob a tecnologia *Eclipse Modeling Framework* (EMF), a ferramenta TMEd [Kudo et al. 2020] oferece suporte ao desenvolvimento de catálogos de padrões em conformidade com o metamodelo SoPaMM.

A criação de um catálogo inicia com uma interface do tipo *wizard*, a partir da qual o usuário consegue definir os elementos que compõem esse catálogo segundo a gramática do SoPaMM. Dentre outras funcionalidades, TMEd permite realizar agrupamentos de padrões relacionados e elaborar padrões de requisitos funcionais e sua estrutura de *features* e cenários, bem como de padrões de teste de aceitação com sua organização de casos e dados de teste.

Por meio dos construtores do metamodelo SoPaMM e do suporte da ferramenta TMEd, um número de CPRS tem sido desenvolvidos pelos autores com integração com padrões de testes. Desse modo, o SoPaMM e a TMEd contribuem não somente com atividades de elicitação e especificação de requisitos, mas também com a atividade de especificação de casos de testes. É por meio desse suporte do SoPaMM e da TMEd na criação de catálogos que integram padrões de requisitos e de testes que esta pesquisa tem desenvolvido um CPRS para S-RES, a partir dos requisitos contidos no manual de certificação da SBIS.

2.3. Manual de Certificação de S-RES

O manual de certificação de S-RES da SBIS reúne requisitos para as categorias de Prontuário Eletrônico do Paciente (PEP) e Telessaúde [Kiatake et al. 2020]. Para a definição desses requisitos, a SBIS compilou experiências e projetos similares de S-RES e baseou-se, em sua maior parte, em normas ISO internacionais. A certificação de S-RES segue um processo de auditoria em que se verifica se o sistema atende a 100% dos requisitos obrigatórios definidos no Manual da Certificação.

Os requisitos para obtenção do certificado da SBIS estão agrupados em três categorias: (i) Estrutura, Conteúdo e Funcionalidades (ECF), requisitos referentes ao fluxo operacional e assistencial, como identificação de pacientes e profissionais; (ii) Nível de Garantia de Segurança 1 (NGS1), requisitos que definem as condições mínimas para uma operação segura e com garantia do sigilo dos dados registrados no sistema; e (iii) Nível de Garantia de Segurança 2 (NGS2), que inclui os requisitos do NGS1 e recursos de certificação digital para o S-RES não operar com registros em papel.

ID	Título	Requisito	Estágio		
			1	2	3
ECF.01 - Identificação de Estabelecimentos de Saúde					
ECF.01.01	Identificação do estabelecimento de saúde	a) O S-RES deve identificar univocamente e registrar o estabelecimento onde está sendo realizada a atenção à saúde específica. b) O cadastro do estabelecimento deve permitir o registro de minimamente os seguintes dados: • Cadastro Nacional de Estabelecimentos de Saúde (CNES); • Cadastro Nacional de Pessoa Jurídica (CNPJ); • Tipo de estabelecimento de saúde (por exemplo, consultório médico, Unidade Básica de Saúde, Unidade de Pronto Atendimento); • Endereço completo do estabelecimento; • Telefone; • e-mail; • Identificação do responsável técnico pelo estabelecimento (vinculado com seu cadastro de profissional).	✓	✓	✓
ECF.01.02	Duplicidade de cadastros de estabelecimentos de saúde	O S-RES deve possuir um mecanismo de validação que emita uma mensagem ao usuário e impeça a continuidade do registro em casos de duplicação de cadastro de estabelecimento de saúde. A validação deve ser realizada pelo menos para o número do CNES e número do CNPJ.	✓	✓	✓

Figura 1. Requisitos de Identificação de Estabelecimento de Saúde, segundo o Manual de Certificação de S-RES [Kiatake et al. 2020]

As categorias ECF, NGS1 e NGS2 apresentam requisitos de estágio 1, 2 e 3. Os de estágio 1 representam o mínimo que todo S-RES deve atender, totalizando 669 requisitos, que são o foco desta pesquisa. Detalhes são encontrados em [Kiatake et al. 2020].

3. Metodologia

Realizou-se um mapeamento sistemático[1] para identificar atividades envolvidas na construção, validação e avaliação de CPRS. A análise dos 25 estudos recuperados constatou uma carência de trabalhos com validação, avaliação, disponibilização e uso prático de CPRS na indústria.

Foram estudados os construtores do SoPaMM e, na prática, como usar a TMEd na construção de CPRS. O aprendizado da TMEd foi feito com CPRS mais simples construídos pelos autores, para registro e autenticação de usuários por meio de *login* e senha. Depois, fez-se um estudo do manual para S-RES, para compreender a organização, a obrigatoriedade e o conteúdo dos requisitos.

Em seguida, deu-se início à construção do CPRS para S-RES com o apoio da ferramenta TMEd. Obedecendo o esquema do metamodelo SoPaMM, foram criados

[1]Material disponível em: http://tiny.cc/xn4vtz.

agrupamentos de padrões para representar os requisitos de estágio 1 das categorias ECF, NGS1 e NGS2. A Figura 1 ilustra a categoria *ECF.01* com requisitos sobre a identificação de estabelecimentos de saúde, os quais foram reunidos em um agrupamento (*SoftwarePatternBag*), chamado *Id_Estabelecimento_ De_Saude*.

Os *requisitos funcionais* de cada categoria são mapeados para um *padrão de requisito funcional*, no qual há a descrição de problema, solução e contexto de uso do padrão. O *comportamento* de um cadastro de um estabelecimento de saúde é mapeado para uma *feature*, na qual descrevem-se *cenários* de cadastro de estabelecimento de saúde em situações de sucesso e insucesso (p.ex. quando o CNPJ ou CNES é inválido ou duplicado).

Perceba na Figura 1 que o requisito ECF.01.01 tornou-se um *FunctionalReqPattern* no trecho do CPRS da Figura 2. Ao mesmo tempo, o requisito ECF.01.02 do manual da SBIS não foi representado como um padrão de requisito funcional, mas sim como cenários do *FunctionalReqPattern* citado. Isto mostra que os requisitos da SBIS necessitam de uma estrutura de organização que facilite não apenas a sua compreensão, mas também o reúso desses requisitos no desenvolvimento e certificação de S-RES.

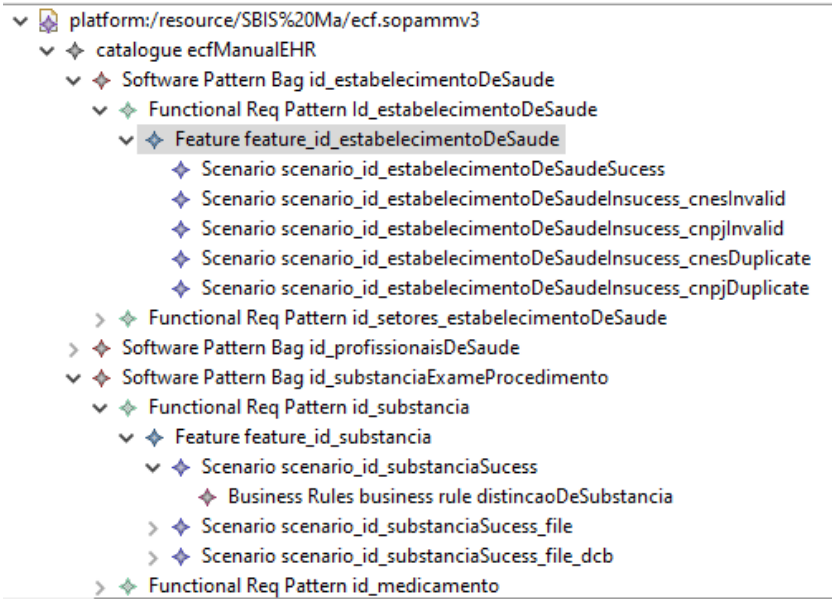

Figura 2. Trecho do CPRS para S-RES na interface da ferramenta *TMed*

Após a definição dos padrões de requisitos funcionais, deve-se elaborar os *padrões de teste de aceitação*, contendo os casos e os dados de teste para os respectivos padrões de requisitos.

4. Método para Avaliar Resultados

O CPRS desenvolvido será validado por especialistas em S-RES, preferencialmente com experiência em Engenharia de Software. Esses participantes deverão analisar: se os requisitos do manual da SBIS foram corretamente representados no CPRS, se nenhum desses requisitos foi desconsiderado, e se os termos empregados no CPRS estão consistentes com os encontrados nos requisitos.

A avaliação do CPRS pode incluir um experimento com a participação de especialistas em S-RES, em que uma simples especificação de requisitos será produzida de duas maneiras: usando apenas o manual da SBIS e com suporte único do CPRS desenvolvido. O objetivo é demonstrar o suporte que o CPRS oferece na especificação de requisitos de S-RES com menor tempo e esforço. Opcionalmente, o CPRS também pode ser avaliado quanto ao suporte à certificação de S-RES, servindo como um *checklist* com requisitos sob uma disposição mais organizada dos requisitos constantes no manual da SBIS.

De modo complementar, pode-se aplicar também um *survey* aos participantes dessas atividades. O objetivo é compreender as percepções dos usuários quanto à utilidade e facilidade de uso do CPRS [Davis 1989].

5. Estado Atual do Trabalho

O CPRS para S-RES está em fase de desenvolvimento. Dos 1318 requisitos de estágio 1, os quais são obrigatórios em toda implementação de S-RES, 599 deles (45%) já foram modelados no CPRS. Ressalta-se que os padrões de teste de aceitação de cada requisito modelado foram definidos e relacionados.

O rastreamento dos padrões de requisitos para os requisitos descritos no manual está disponível para consulta[2]. Planeja-se finalizar a modelagem do catálogo após o exame de qualificação, que ocorrerá em 18/06/21. Esta pesquisa de mestrado tem previsão de defesa em 23/02/22.

6. Contribuições Esperadas

- Um catálogo público de padrões de requisitos para S-RES, validado e avaliado por especialistas dessa classe de sistemas;
- Disseminação da prática de reúso por meio de padrões de requisitos.

7. Comparação com Trabalhos Relacionados

O mapeamento sistemático realizado recuperou 25 estudos que envolvem CPRS, dentre os quais apenas sete realizam validação, e três, algum tipo de avaliação. Em nenhum estudo houve relato de manutenção de CPRS, talvez por falta de ferramenta de suporte. Cerca de 60% dos CPRS encontrados não está disponível para acesso, e apenas 20% dos CPRS estão públicos na Web. Além disso, apenas três dos 25 estudos apresentam uso prático de CPRS na indústria.

Dadas as lacunas apresentadas, o desenvolvimento e a divulgação de um CPRS para um novo domínio de aplicação, ainda não explorado, como o de S-RES, tem potencial para disseminar a prática de reúso via padrões de requisitos. Com a validação e avaliação por especialistas em S-RES, aumentam as chances de o CPRS ser utilizado pela indústria de software, algo escasso como apuramos.

8. Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de

[2]Material disponível em: http://tiny.cc/5n72uz.

Pessoal de Nível Superior - Brasil (CAPES) – bolsa de pesquisa CAPES DS.

Referências

Amorndettawin, M. and Senivongse, T. (2019). Non-functional requirement patterns for agile software development. In *Proceedings of the 2019 3rd International Conference on Software and E-Business*, ICSEB 2019, p. 66–74, Association for Computing Machinery, New York, NY, USA.

Beckers, K., Côté, I. and Goeke, L. (2014). A catalog of security requirements patterns for the domain of cloud computing systems. In *Proceedings of the ACM Symposium on Applied Computing*, p. 337-342, Gyeongju, Republic of Korea.

Chelimsky, D., Astels, D., Helmkamp, B., North, D., Dennis, Z. and Hellesoy, A. (2010), The RSpec Book: Behaviour Driven Development with Rspec, Cucumber, and Friends, Pragmatic Bookshelf, Raleigh, NC, 1st edition.

Chernak, Y. (2012). Requirements reuse: The state of the practice. In *2012 IEEE International Conference on Software Science, Technology and Engineering*, SWSTE 2012, Herzlia, Israel, June 12-13, p. 46–53, IEEE Computer Society, Los Alamitos, CA, USA.

Costal, D., Franch, X., López, L., Palomares, C. and Quer, C. (2019). On the use of requirement patterns to analyse request for proposal documents. In *Proceedings of the 38th International Conference on Conceptual Modeling*, ER 2019, Salvador, Brazil, November 4-7, p. 549-557, Springer.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly* 13(3), pages 319-340.

Irshad, M., Petersen, K. and Poulding, S. (2018). A systematic literature review of software requirements reuse approaches. *Inf. Softw. Technol.* 93(C), pages 223–245.

Kiatake, L. G. G. et al. (2020) "Manual de certificação de sistemas de registro eletrônico em saúde", Sociedade Brasileira de Informática em Saúde. Versão 5.0. Novembro de 2020. Instituído e regido pela Resolução CFM nº 1821/2007.

Kopczynska, S., Nawrocki, J. R. and Ochodek, M. (2018). An empirical study on catalog of non-functional requirement templates: Usefulness and maintenance issues. *Inf. Softw. Technol.* 103, pages 75-91.

Kudo, T. N., Bulcão-Neto, R. F. and Vincenzi, A. M. R. (2020). Requirement patterns: A tertiary study and a research agenda. *IET Software* 14(1), pages 18-26.

Kudo, T. N., Bulcão-Neto, R. F. and Vincenzi, A. M. R. (2019). A conceptual metamodel to bridging requirement patterns to test patterns. In *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*, p. 155-160, ACM, New York, NY, USA.

Kudo, T. N., Bulcão-Neto, R. F. and Vincenzi, A. M. R. (2020). Uma ferramenta para construção de catálogos de padrões de requisitos com comportamento. In *Anais do Workshop em Engenharia de Requisitos*, p. 1-14, Editora PUC-Rio.

Palomares, C., Quer, C. and Franch, X. (2014). Requirements reuse with the PABRE framework. *Requirements engineering magazine* 2014, p. 1-10.

Palomares, C., Quer, C., Franch, X., Renault, S. and Guerlain, C. (2013) A catalogue of functional software requirement patterns for the domain of content management systems. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC '13, Coimbra, Portugal, March 18-22, 2013. p. 1260-1265.

Palomares, C., Quer, C., Franch, X., Guerlain, C. and Renault, S. (2012). A catalogue of non-technical requirement patterns. In *2012 Second IEEE International Workshop on Requirements Patterns (RePa)*, p. 1-6.

Riaz, M., Elder, S. and Williams, L.A. (2016). Systematically developing prevention, detection, and response patterns for security requirements. In *24th IEEE International Requirements Engineering Conference*, p. 62-67, IEEE Computer Society.

Tockey, S. (2015). Insanity, hiring, and the software industry. *Computer* 48, pages 96–101.

Withall, S. (2007), Software Requirement Patterns. Best practices, Microsoft Press, Redmond, Washington.