

ESQUEMA E EDITOR DE METADADOS BASEADO NO DUBLIN CORE PARA EXPERIMENTOS CONTROLADOS DE ENGENHARIA DE SOFTWARE

Filipe Amadeu Santana (PIBIC/CNPq/FA/UEM), Edson Alves de Oliveira Junior (Orientador), André Felipe Ribeiro Cordeiro (Coorientador). E-mail: edson@din.uem.br, afrcordeiro2@uem.br.

Universidade Estadual de Maringá, Centro de Tecnologia, Departamento de Informática, Maringá, PR.

Área e subárea do conhecimento: Ciência da Computação, Engenharia de Software.

Palavras-chave: Engenharia de Software Experimental, Metadados, Perfil de Metadados

RESUMO

Este trabalho apresenta o desenvolvimento de um perfil de aplicação de metadados e de um editor baseado no padrão Dublin Core (DC) para experimentos controlados de Engenharia de Software (ES). O objetivo foi propor uma solução que padronize o registro de artefatos experimentais. A metodologia envolveu revisão da literatura sobre esquemas e perfis de aplicação, resultando na análise de 29 perfis e 12 esquemas. O AGRIS foi adotado como referência por sua documentação estruturada. O perfil, denominado DCEP-SE, foi construído em etapas iterativas: na versão inicial, todos os elementos do DC foram aplicados; em seguida, elementos representando as fases de Wohlin et al. foram incorporados; por fim, o estudo de Porter et al. foi utilizado para popular o perfil, removendo progressivamente elementos até identificar um conjunto mínimo essencial. A estrutura final compreende subject.team, subject.scope, subject.planning, subject.operation, subject.analysis e subject.presentation, complementados por qualificadores em description.*. Em seguida, foi implementado um editor em HTML, CSS e JavaScript, sem banco de dados, baseado em casos de uso e histórias de usuário. Foram definidos dois campos obrigatórios (Título e Data), com até cinco instâncias adicionais por elemento e possibilidade de remoção dos opcionais. A avaliação inicial, baseada no modelo TAM, apontou percepções positivas quanto à simplicidade e utilidade, com sugestões de melhorias de usabilidade e integração. Conclui-se que os objetivos foram atingidos e prevê-se, como trabalho futuro, a extensão do perfil para apoiar reprodutibilidade.

INTRODUÇÃO

Experimentos em ES geram diversos artefatos, como código-fonte, modelos, especificações e dados (ABOU KHALIL; ZACCHIROLI, 2022). O gerenciamento desses artefatos é dificultado pela ausência de padronização, comprometendo replicabilidade e reuso (TIMPERLEY et al., 2021). Metadados, entendidos como

dados sobre dados, quando organizados em padrões, auxiliam na padronização e na interoperabilidade (SANTANA; CORDEIRO; OLIVEIRAJR, 2023). Entretanto, estudos anteriores apontam que padrões genéricos não capturam adequadamente características de experimentos, resultando em ambiguidades ou impossibilidade de representar determinados elementos. Diante disso, surge a necessidade de adotar perfis de aplicação de metadados específicos para a Engenharia de Software Experimental (ESE). Um perfil de aplicação integra termos selecionados de diferentes vocabulários, incluindo restrições, para atender necessidades de um contexto particular (MIKSA *et al.*, 2021). Este projeto tem como objetivo desenvolver um perfil de aplicação e um editor de metadados baseados no padrão DC para experimentos controlados de ES.

MATERIAIS E MÉTODOS

O desenvolvimento do DCEP-SE iniciou-se com uma revisão de literatura para identificar perfis existentes. Foram encontrados 29 perfis de aplicação e 12 esquemas de metadados. Os esquemas foram descartados por apresentarem estrutura rígida e pouco adaptável a experimentos. Entre os perfis, o AGRIS foi selecionado como referência, devido à sua documentação detalhada e sistemática.

Com base nessa análise, o processo de construção do perfil foi conduzido em etapas iterativas. Na versão inicial, todos os 15 elementos do DC foram associados às fases experimentais, resultando em redundâncias. Em seguida, elementos representando as etapas do processo experimental de Wohlin et al. (2024) foram introduzidos, com aplicação sequencial dos termos do DC. Na terceira versão, o elemento dc:description passou a centralizar a representação, mapeando cada fase de forma mais concisa. O estudo de Porter et al. (1995) foi utilizado para popular o perfil, e os elementos foram removidos progressivamente até que qualquer nova exclusão comprometesse a coerência da descrição.

A segunda etapa consistiu na implementação de um editor para registro estruturado dos elementos definidos no perfil. O desenvolvimento foi orientado por histórias de usuário, requisitos funcionais e não funcionais, além de modelagem de casos de uso. Foram definidos dois campos obrigatórios: Título e Data. Essa decisão foi adotada de forma pragmática, pois não foi possível realizar um estudo formal para decidir, de maneira empírica, quais campos seriam obrigatórios ou opcionais. Para garantir consistência mínima e viabilizar a implementação, definiu-se também que cada elemento poderia conter até cinco instâncias adicionais e que todos os campos, com exceção dos obrigatórios, poderiam ser removidos.

Os casos de uso principais foram: Inserir Metadados (C1), Manipular Campos de Metadados (C2) e Exportar Metadados (C3). Os diagramas de caso de uso e de classes foram elaborados na ferramenta ASTAH. A implementação utilizou apenas HTML, CSS e JavaScript, dispensando frameworks modernos e banco de dados, uma vez que não havia necessidade de autenticação nem armazenamento persistente.

RESULTADOS E DISCUSSÃO

O perfil reflete as fases do processo experimental de Wohlin et al. (2024), garantindo representação mínima para documentação. Informações gerais, como título, equipe, autores e contribuidores, ficam em `subject.team`; problema e objetivos, em `subject.scope`; planejamento, em `subject.planning`, incluindo contexto, hipóteses, variáveis, amostra, design, instrumentos e ameaças à validade; execução, em `subject.operation`, com preparação, protocolo, validação e cobertura geográfica; análise, em `subject.analysis`, com estatísticas, redução de dados e testes de hipóteses; disseminação, em `subject.presentation`, com referência bibliográfica, pacote experimental, DOI, formato, idioma, fonte e direitos autorais. Qualificadores `description.*` detalham hipóteses, variáveis, resultados e demais informações. Essa organização, construída a partir de iterações sucessivas e exclusões progressivas de elementos, garante clareza, consistência e evita redundâncias, permitindo que experimentos distintos sejam documentados de forma comparável e interoperável.

O editor implementa os elementos do DCEP-SE, possibilitando inserção, remoção e exportação de campos. Os campos obrigatórios de título e data asseguram consistência mínima. Todos os demais campos podem ser removidos, e cada elemento permite até cinco instâncias adicionais. O sistema valida obrigatórios, exibe mensagens de erro em casos de inconsistência e disponibiliza ajuda contextual. A exportação gera automaticamente arquivo estruturado, compatível com intercâmbio em diferentes contextos.

A arquitetura baseada em HTML, CSS e JavaScript garantiu compatibilidade com navegadores modernos, responsividade e simplicidade de uso. A ausência de frameworks adicionais ou banco de dados contribuiu para desempenho adequado, de acordo com os objetivos do projeto.

A avaliação do editor foi realizada com base no modelo Technology Acceptance Model (TAM). O questionário contempla Utilidade Percebida (PU), Facilidade de Uso Percebida (PE) e Intenção de Uso (IU). Resultados iniciais apontam percepções positivas quanto à simplicidade da interface e clareza da interação. Usuários destacaram a facilidade de aprendizagem, a responsividade e a utilidade no registro de metadados. Como limitações, foram mencionadas a ausência de integração com ferramentas externas e a necessidade de melhorias de usabilidade, como atalhos de teclado e maior aplicação de conceitos de UX. Sugestões incluíram integração futura com repositórios digitais e ajustes na interface.

CONCLUSÕES

O projeto desenvolveu o perfil de metadados DCEP-SE para experimentos controlados de ES e um editor para sua utilização. O perfil foi construído iterativamente, e o editor implementado com funcionalidades de manipulação e exportação de metadados.

A avaliação inicial indicou percepção positiva sobre utilidade e facilidade de uso, com sugestões de melhorias. Os objetivos geral e específicos foram atendidos.

Como trabalho futuro, está em desenvolvimento um perfil voltado à reprodutibilidade de experimentos, que será incorporado ao editor.

AGRADECIMENTOS

Os autores agradecem o apoio financeiro do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq). O apoio do CNPq por meio da bolsa de estudos foi essencial para a realização deste projeto e é profundamente valorizado e apreciado.

REFERÊNCIAS

ABOU KHALIL, Z.; ZACCHIROLI, S. Software artifact mining in software engineering conferences: A meta-analysis. In: **Proceedings of the 16th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement – ESEM '22**, p. 227–237. New York, NY, USA: Association for Computing Machinery, 2022.

MIKSA, T.; SIMMS, S.; MIETCHEN, D.; JONES, S.; NEISH, P. et al. Using application profiles to explore domain-specific metadata for data management plans. **Data Science Journal**, v. 20, p. 1–17, 2021.

PORTER, A. A.; VOTTA, L. G.; BASILI, V. R.; ARDIS, M.; KELLY, J.; WEISS, D.; GANNON, J.; GERBER, R.; LOADER, C.; SLUD, E.; VANDERWEIL, S. **Comparing detection methods for software requirements inspections: a replicated experiment**. *IEEE Transactions on Software Engineering*, v. 21, n. 6, p. 563–575, 1995.

SANTANA, F.; CORDEIRO, A.; OLIVEIRAJR, E. Use of the Dublin Core Standard to Express Open Metadata Related to Software Engineering Experiments. In **WORKSHOP DE PRÁTICAS DE CIÊNCIA ABERTA PARA ENGENHARIA DE SOFTWARE, III**, 2023, Campo Grande, MS. **Anais [...]**. Porto Alegre: SBC, 2023. p. 1-5. DOI: 10.5753/opensciense.2023.235672.

TIMPERLEY, C. S.; HERCKIS, L.; DE GOUE, C. L. et al. Understanding and improving artifact sharing in software engineering research. **Empirical Software Engineering**, v. 26, n. 67, 2021.

WOHLIN, C.; RUNESON, P.; HÖST, M.; OHLSSON, M. C.; REGNELL, B.; WESSLÉN, A. **Experimentation in Software Engineering**. Springer, 2024.