

Guia interativo de boas práticas em Engenharia de Software

Jonathas A. Almeida^[1], Helder J. L. Rangel^[2], Matheus A. C. Costa^[3], Maurício B. L. Filho^[4], Francisco P. A. Medeiros^[5], Heremita B. Lira^[6]

[1] jonathasarts@gmail.com. [2] helder.rangel@academico.ifpb.edu.br. [3] matheus.augusto.jp@gmail.com.

[4] mauricio.lira@academico.ifpb.edu.br. [5] petronio@ifpb.edu.br. [6] heremita@ifpb.edu.br. IFPB Campus João Pessoa.

RESUMO

Ao analisar a diversidade de boas práticas da Engenharia de Software presentes no mercado atual, o Grupo de Pesquisa de Engenharia de Software do Instituto Federal da Paraíba observou a necessidade de reunir e apresentar estas boas práticas em um guia web centralizador, que irá facilitar o acesso, a identificação e a aplicação adequada de tais práticas em diversos cenários de projetos de software. O objetivo deste trabalho é apresentar os resultados do desenvolvimento de uma primeira versão funcional deste guia web, para disponibilização de boas práticas em Engenharia de Software pesquisadas pelo grupo. O projeto é desenvolvido utilizando metodologias ágeis e é fruto de um conjunto de outros projetos complementares e colaborativos do grupo de pesquisa: Repositório de Práticas em Gerenciamento de Projeto de Software e Repositório de Práticas em Processo de Desenvolvimento de Software, conjuntamente com o projeto de pesquisa Ambiente de Boas Práticas Reconfiguráveis em Engenharia de Software.

Palavras-chave: Processo de Desenvolvimento de Software. Engenharia de Software. Boas Práticas. Metodologia Ágil. Guia Web.

ABSTRACT

Analyzing the diversity of best practices in Software Engineering present in industry, the Software Engineering Research Group of the Federal Institute of Paraíba realized the need to organize them together in a web repository so that the access, identification and proper application in various software project scenarios is easier. The objective of work is to present the first functional version of this web environment, in order to enable the best practices in Software Engineering as researched by the group. This project has been developed using agile methodologies and is the result of a trio of complementary and collaborative projects from the research group: Repository of Practices in Management of Software Project, Repository of Practices in the Software Development Process, together with the project Reconfigurable Best Practices Environment in Software Engineering.

Keywords: *Software Development Process. Software Engineering. Best Practices. Agile Methodology. Web Environment.*

1 Introdução

O cenário mercadológico atual faz com que, cada vez mais, as organizações invistam em práticas inovadoras que minimizem custos, maximizem o retorno de investimentos e satisfaçam aos clientes. Na indústria de software, este cenário não é diferente. As sugestões de boas práticas advindas da Engenharia de Software (ES) são fatores de sucesso técnico associado aos seus serviços, processos e produtos. Nesse contexto, as metodologias de gerenciamento e desenvolvimento ágil de projetos de software ganham cada vez mais espaço dentro das organizações (AMBLER, 2016).

A ES lida com metodologias, ferramentas e disciplinas que colaboram para o desenvolvimento de sistemas computacionais. Ela aprimora a computação, utilizando metodologias de Gerenciamento de Projetos (GP) integrada ao Processo de Desenvolvimento de Software (PDS) (SOARES, 2015). Para garantir a geração de produtos de qualidade, as empresas devem sistematizar as suas atividades, definindo seus PDS e métodos de GP que atendam às suas características, considerando particularidades de projetos, equipes e clientes (RODRIGUES, 2013).

Diante deste cenário e das mais diversas e numerosas técnicas para o gerenciamento e desenvolvimento ágil de projetos de software disponíveis, existe a problemática de identificar e aplicar adequadamente as boas práticas da ES para cenários específicos de projetos de software no setor produtivo.

Tendo em vista a ampla variedade de práticas, a boa quantidade de informação e a pouca centralização para se encontrarem as boas práticas de ES, o projeto "Ambiente de Boas Práticas Reconfiguráveis em Engenharia de Software" (ABPRES) (ALMEIDA, S. *et al.*, 2017; ALMEIDA, J. *et al.*, 2017) visa desenvolver um guia web para disponibilizar, em um local centralizado, as boas práticas de ES. Espera-se que o sistema web, quando for implantado, possibilite o conhecimento e troca de informações entre desenvolvedores, empresas e acadêmicos sobre as formas de aplicação, utilização e melhorias das boas práticas mais utilizadas no mercado, tornando-se uma referência para as encontrar e as discutir.

2 Referencial teórico

O software é o conjunto de vários artefatos e não apenas o código fonte (SOMMERVILLE, 2007).

Segundo Pressman (2006), um software é um conjunto composto por instruções de computador, estruturas de dados e documentos.

A Engenharia de Software é uma abordagem sistemática e disciplinada para o desenvolvimento de software (PRESSMAN, 2006). Para Sommerville (2007), a engenharia de software é uma disciplina de engenharia relacionada com todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até sua manutenção, já em operação.

Um dos grandes desafios da Engenharia do Software é a satisfação do cliente com relação ao software produzido (JALOTE, 2005). Nesse sentido, o conhecimento das boas práticas de Gerenciamento de Projetos de Software e dos seus respectivos cenários de aplicação deverá contribuir para que os resultados dos projetos de software se aproximem dos objetivos traçados.

Um projeto é um conjunto de atividades temporárias, realizadas em grupo, destinadas a produzir um produto, serviço ou resultado únicos. O Gerenciamento de Projetos é a aplicação de conhecimentos, habilidades e técnicas para a execução de projetos de forma efetiva e eficaz. Trata-se de uma competência estratégica para organizações, permitindo que elas unam os resultados dos projetos com os objetivos do negócio para melhor competir em seus mercados (PMI, 2014).

Assim como qualquer processo produtivo, a produção de software necessita de uma sistematização que possa definir os passos, as atividades que serão realizadas, a dependência entre essas atividades, as suas entradas e saídas, os papéis e as responsabilidades de cada um dos atores envolvidos no desenvolvimento de software (RODRIGUES; ESTRELA, 2012).

3 Método de pesquisa

De acordo com Prodanov e Freitas (2013), considerando a metodologia de pesquisa científica, este projeto é de Natureza Aplicada, pois procura gerar conhecimentos para aplicação prática direcionados à solução de problemas específicos. Considerando o objetivo do estudo, pode-se classificar este projeto de pesquisa como Exploratório, pois ele possibilita maior familiaridade com o problema, tornando-o explícito. A abordagem de pesquisa adotada é quali-quantitativa e os principais procedimentos técnicos utilizados são: pesquisa bibliográfica, levantamento e estudo de caso.

Do ponto de vista de execução, para o desenvolvimento do ABPRES, foram necessários o estudo e aplicação de alguns conceitos relacionados ao desenvolvimento ágil de software, como: adequação às mudanças, equipes auto-organizadas, entregas periódicas de software funcional, uma vez que o sistema desenvolvido deva ser simples e apresentar o conteúdo de forma organizada e sucinta, para que possa ser utilizado tanto por profissionais como por estudantes na área de TI (Tecnologia da Informação).

Nosso estudo foi desenvolvido nas seguintes etapas: Levantamento de Informações, Estratificação das Informações, Desenvolvimento do Guia Web para Disponibilização das Boas Práticas e Qualificação Continuada dos Pesquisadores. O desenvolvimento do guia web utiliza uma adaptação das metodologias ágeis XP e SCRUM (HENRIK, 2007) para desenvolvimento e gerenciamento de software, que consta de etapas iterativas e incrementais para entregas parciais.

Este projeto vem sendo executado de forma iterativa e incremental, utilizando um Framework de Gerenciamento Ágil de Projetos de Software (MORAIS *et al.*, 2017a) e um Processo de Desenvolvimento Ágil de Software (ALMEIDA, S. *et al.*, 2018), definidos, respectivamente, em projetos colaborativos do GPES, o Repositório de Práticas em Gerenciamento de Projeto de Software (MORAIS *et al.*, 2017b) e o Repositório de Práticas em Processo de Desenvolvimento de Software (SILVA *et al.*, 2017).

Partindo do levantamento de requisitos, escreveram-se as *User Stories* com as Mensagens e Regras de Negócio associadas. Assim, os documentos de Regras de Negócio e Mensagens são referenciados pelas respectivas US.

Os próximos passos foram a elaboração, ou revisão, a partir da segunda *Sprint*, do(s): 1. Design de Arquitetura; 2. Diagramas de Classe; 3. Diagramas de Sequência, caso existam; 4. Protótipos. Os protótipos foram elaborados a partir da US e Diagramas de Sequência, e, com todos estes artefatos, a equipe de desenvolvimento, responsável pelos testes, construiu os Casos de Teste, considerando os objetos, os dados envolvidos e o Protótipo da interface visual do sistema.

A elaboração dos casos de teste ocorreu em paralelo à elaboração da interface visual e criação das *views*. Para melhor implementação dos Protótipos da interface na criação das *views*, tornando-os mais amigáveis e concisos para os usuários, a equipe conta com a participação de um designer gráfico, que tem trabalhado na interface do sistema.

Na codificação das tarefas, para refinamento do código, fez-se uso de testes unitários em apenas alguns dos requisitos, devido ao pouco tempo disponível. Feita a codificação, a equipe de desenvolvimento ficou responsável pelo versionamento do sistema, para realização dos Testes Automatizados, elaborados por parte da equipe encarregada dos testes. Realizada execução dos Testes Automatizados, reportaram-se os erros e correções necessárias, documentando-os sempre, para que os programadores realizassem os devidos ajustes, repetindo, então, este ciclo de teste e correções até a homologação do produto.

Os procedimentos utilizados para a pesquisa bibliográfica consistiram basicamente na realização de busca por artigos, periódicos, livros e sistemas e sites na internet relacionados à utilização e disponibilização de boas práticas da ES.

Em todo o desenvolvimento, utilizou-se a linguagem de programação PHP (*Hypertext Preprocessor*), na versão 7.2.0, com a base de dados construída a partir do Sistema de Gerenciamento de Banco de Dados – SGBD PostgreSQL 9.4.14. Segundo Heuser (1998), um sistema de gerenciamento de banco de dados é um software que incorpora as funções de definição, recuperação e alteração de dados em um banco específico. Para acessar um banco de dados, é necessário possuir um driver, que nada mais é do que um tipo de software que tem por objetivo permitir a comunicação entre dois sistemas – o driver utilizado foi o PostgreSQL ODBC Driver.

O PostgreSQL foi escolhido por ser um dos bancos de dados objeto-relacional de código aberto que mais se destacam no mercado, sendo um dos poucos que suportam a complexidade de um sistema para Web. Ele possui recursos essenciais para o bom funcionamento de um sistema dessa complexidade como: Suporte a consultas complexas, Integridade referencial, Suporte ao SQL92 e SQL99, Suporte a grande quantidade de dados, Integridade transacional.

A extensão PDO (*PHP Data Objects*) define uma interface leve e consistente para acessar bancos de dados em PHP. Cada driver de banco de dados que implementa a interface PDO pode expor recursos específicos do banco de dados como funções de extensão regulares. Essa extensão fornece ainda uma camada de abstração de acesso a dados. Por não levar em consideração as sintaxes específicas das bases de dados, permite que o processo de mudança de bases de dados e plataformas seja, praticamente, sem problemas, simplesmente mudando os dados de

conexão. A Figura 1 mostra a arquitetura da extensão PDO e como ela se conecta ao driver e este à base de dados.

O PHP oferece recursos que são utilizados em diversos paradigmas de programação como: (1) Programação estruturada – apesar de recomendado não se usa este paradigma atualmente, a linguagem oferece o suporte para ser usado, caso necessário; (2) Programação Funcional – funções anônimas com suporte a *closure*, *bind* de funções a variáveis, recursividade, entre outras; (3) Programação Orientada a Objetos – *Namespace*, *Classes*, *Classes Abstratas*, *Interfaces*, *Clonagem*, *Herança*, *Exceptions*, *Constructors* e *Destructors*, entre outras; (4) Metaprogramação – API de Reflexão e métodos mágicos como `__toString()`, `__sleep()`, `__wakeup()`, `__clone()`, `__invoke()`, `__set()`, `__get()`, entre outras.

Utilizou-se como base para o desenvolvimento o framework SLIM PHP, que, assim como a maioria dos *frameworks Web*, usa alguma variação do padrão de projeto MVC (*Model View Controller*) em que a camada da visão (*view*) é a responsável pela apresentação. A *view* recebe o estado do modelo através do controlador. Os objetos dessa camada recebem os dados de entrada do usuário que serão repassados para o controlador (FOX, 2006). A camada do controlador (*controller*) interliga a camada da visão à camada de negócio ou modelo, ou seja, é a fronteira entre ambas as camadas. Por fim, o modelo (*model*), camada que abriga a verdadeira lógica de negócio. O modelo contém as regras para obtenção e atualização do estado e fornece ao controlador o acesso aos dados, pois é a única parte do sistema que se comunica com o banco de dados (BASHAM *et al.*, 2008).

Outra facilidade apresentada pelo PHP é a presença do Composer – um gerenciador de dependências ou pacotes para PHP que vem ganhando espaço e se tornando cada vez mais indispensável no uso da linguagem –, o que favorece o desenvolvimento colaborativo.

Durante o processo de planejamento, uma etapa muito importante foi a de levantamento de requisitos. Nessa atividade, os membros da equipe de desenvolvimento, em conjunto com membros alocados aos outros projetos do GPES, trabalham em reuniões semanais, para definir o domínio da aplicação, quais serviços o sistema deveria fornecer, modelos de interface, as restrições de hardware, entre outros.

Nessa fase, foi criado o Documento de Requisitos, o qual, segundo Pádua (2001), é o documento oficial

de descrição dos requisitos de um projeto de software. O Documento de Requisitos adotado como padrão para especificar o sistema apresenta as seguintes características: (1) Funcionalidade: O que o software deverá fazer? (2) Interfaces externas: Como o software interage com as pessoas, com o hardware do sistema, com outros sistemas e com outros produtos? (3) Padrões: Como o software deverá ser desenvolvido? (4) Confiabilidade: O que pode se esperar do sistema, tratando-se de segurança, consistência?

Após ter definido o escopo do sistema e este ter sido aprovado pelos membros e coordenadores do GPES, iniciou-se a fase de definição da arquitetura, definição das tecnologias a serem utilizadas e a elicitação e análise de requisitos.

A especificação dos requisitos foi feita de forma colaborativa – em um documento compartilhado, os membros participantes descreveram os requisitos para uma posterior análise do grupo a partir de discussões, fazendo-se as modificações e ajustes necessários. A equipe de desenvolvimento conta com analistas, desenvolvedores, gerentes, testadores, designer e orientadores.

A Garantia de Qualidade de Software (SQA) é um fator muito importante para qualquer empresa, é o padrão de planejamento e ações estabelecidas para garantir a qualidade do produto (PRESSMAN, 1995). A atividade de teste, uma das funções do grupo SQA, reúne um grande pacote de casos de testes que ajudam a detectar erros. Uma atividade de testes cuidadosa revela a maioria dos erros após desenvolvimento do produto.

Para a realização de testes automatizados, a equipe utilizou o Selenium IDE (*Integrated Development Environment*), ferramenta utilizada para o desenvolvimento de *scripts* de teste com o Selenium através de um *plugin* para o navegador web Firefox, que torna o desenvolvimento dos *scripts* mais eficiente pelo método *Record and Replay* (Gravação e Execução).

4 Resultados

O principal resultado do projeto foi a produção de um protótipo funcional do guia proposto, dando uma visão mais ampla para que o grupo GPES pudesse aprimorar, cada vez mais, a forma de apresentação das boas práticas em ES e as funcionalidades necessárias para garantir uma participação colaborativa dos profissionais e acadêmicos que convivem diariamente com a aplicação dessas boas práticas.

Das funcionalidades implementadas no protótipo, destaca-se a de buscar boas práticas, como apresentada a seguir. Na página inicial do sistema, mostrada na Figura 1, é apresentada uma barra de pesquisa, com a possibilidade de se utilizarem filtros, tais como, categorias e tags do conteúdo a ser buscado.

Figura 1 – Tela Inicial



Fonte: Autoria própria.

O usuário cadastrado no sistema pode entrar com sua conta para um maior interação, na opção de login – para criar uma nova conta, basta selecionar a opção “Criar conta” (ver Figura 2).

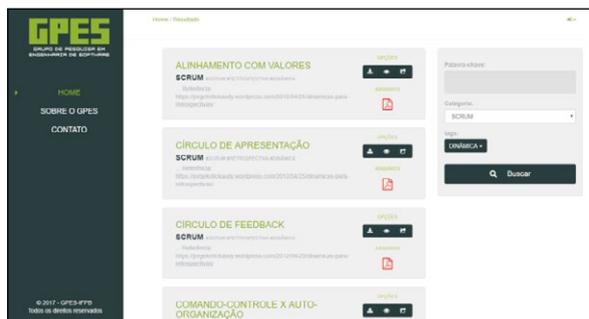
Figura 2 – Tela Inicial – Detalhe Criar Conta



Fonte: Autoria própria.

Ao buscar as boas práticas da categoria “SCRUM”, por exemplo, temos o resultado apresentado na Figura 3, com as boas práticas encontradas.

Figura 3 – Tela Resultado da Busca



Fonte: Autoria própria.

Na tela mostrada na Figura 3, o usuário pode refinar a pesquisa realizada nas opções apresentadas à direita, na Figura 4. Os campos aparecem preenchidos conforme a busca realizada.

Figura 4 – Detalhe Refinar Busca



Fonte: Autoria própria.

É possível, ainda, ver mais detalhes ou realizar o download de uma boa prática selecionada na lista, ao centro da tela. A Figura 5 mostra como as informações são apresentadas.

Figura 5 – Campos da Boa Prática



Fonte: Autoria própria.

Algumas informações básicas de cada boa prática do resultado são apresentadas para facilitar sua identificação: (A) título, (B) categoria, (C) tags, (D) um breve resumo e, ainda, (E) os tipos de arquivo associados.

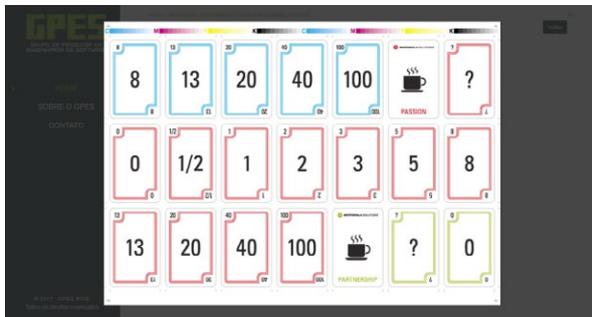
Figura 6 – Tela Detalhes Boa Prática



Fonte: Autoria própria.

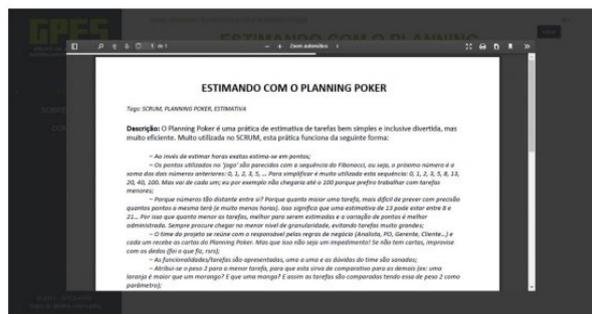
Ao visualizar os detalhes de uma boa prática na Figura 6, existe ainda a possibilidade de download de arquivos, um ou mais, associados à boa prática, sem a necessidade do download de toda ela. A visualização dos detalhes dos arquivos é apresentada nas Figuras 7 e 8.

Figura 7 – Detalhe Visualização de Imagem



Fonte: Autoria própria.

Figura 8 – Detalhe Visualização de PDF



Fonte: Autoria própria.

Outras informações que serão adicionadas nesta tela são uma lista de boas práticas associadas e uma lista de comentários/observações de outros usuários sobre o uso da boa prática em exibição. Para um usuário logado no sistema, existirá ainda a possibilidade de adicionar um novo comentário/observação, interagindo com outros colaboradores.

Os administradores do ABPRES serão responsáveis pela inserção e atualização do conteúdo pesquisado e organizado pelo GPES no sistema. Ao entrar com o perfil de administrador, o usuário será direcionado ao painel administrativo do sistema, onde poderá visualizar as listas de boas práticas e categorias e usuários cadastrados. O processo de Cadastro de Conteúdo é uma operação válida apenas para o usuário administrador.

Ao realizar o cadastro de conteúdo, o usuário administrador, caso insira dados inválidos ou deixe de preencher um dos campos obrigatórios, será informado pelo sistema para que possa realizar as devidas correções (ver Figura 9).

Figura 9 – Detalhe Mensagem de Alerta



Fonte: Autoria própria.

Caso o cadastro seja efetuado com sucesso, o sistema apresentará uma mensagem de confirmação da operação, como pode ser visto na Figura 10.

Figura 10 – Detalhe Mensagem de Confirmação



Fonte: Autoria própria.

Durante a pesquisa sobre os softwares de gerenciamento de conteúdo disponíveis, verificou-se a necessidade de um repositório digital onde devem ser armazenadas o material coletado e organizado

sobre as boas práticas, assim como um gerenciador deste repositório que possa garantir a preservação e visibilidade desse material.

As reuniões presenciais e remotas possibilitaram uma melhor compreensão das características do guia web que está em desenvolvimento. Algumas necessidades para o desenvolvimento do guia foram levantadas como, por exemplo, um endereço válido na rede do IFPB onde o sistema deve funcionar. O guia vai requerer usabilidade, facilidade de uso, intuição, interface amigável e pouca ou nenhuma complexidade.

A partir do PDS elaborado no projeto Repositório de Práticas em Processos de Desenvolvimento de Software (RPPDS), definiram-se os papéis dos participantes para melhor desenvolvimento das atividades.

5 Conclusão

Ao avaliar o protótipo, verificou-se que a apresentação das informações referentes às boas práticas deve ser dinâmica, clara e objetiva. Assim, os usuários compreenderão melhor e poderão se envolver com os dados apresentados. Outro ponto levantado foi a possibilidade de criação de um vocabulário controlado para melhor indexação das informações. A utilização do vocabulário trará grandes melhorias na pesquisa, por assunto, dentro do repositório, proporcionando uma recuperação da informação de forma mais eficiente.

Baseado em uma avaliação ad-hoc realizada pelos membros do projeto e por alguns desenvolvedores de software convidados, concluiu-se ser necessária a continuidade do desenvolvimento de novas versões do ABPRES – um dos requisitos indispensáveis é a construção de uma interface mais envolvente para apresentar um conteúdo relevante, tanto em termos de interatividade quanto de recursos visuais.

Com isso, o guia poderá se tornar mais rico ao apresentar vídeos, animações, imagens etc., ao invés de apresentar apenas textos. O grupo avalia a possibilidade de utilizar vídeos para demonstrar a aplicação de boas histórias, por exemplo, usando o *storytelling*, ou seja, contar uma boa história para despertar o interesse do usuário e que, de preferência, marque-o, fique em sua memória. Uma narrativa bem articulada, com começo, desenvolvimento e final específicos e que, de alguma forma, o capture (ENDEAVOR BRASIL, 2015).

Da mesma forma, o grupo pretende disponibilizar mapas mentais, apresentando as informações

associadas com palavras e imagens, recurso que, além de ser mais atrativo, ajuda a reter os dados com mais agilidade e eficiência, usando o potencial criativo da mente (BUZAN, 2009). Os mapas poderão ser baixados para que o usuário os utilize como ferramenta de comunicação das boas práticas encontradas no guia.

A parceria com a equipe de designers do curso de Design Gráfico poderá ainda viabilizar a produção de conteúdo em vídeo bem como ilustrações para enriquecimento da comunicação do ABPRES. A metodologia e o produto gerado neste trabalho permitirão ao GPES avançar no processo de catalogação e disponibilização das boas práticas para todos bem como a forma de aplicação dessas práticas para diversos tipos de projetos.

Por último, é necessário reforçar que a utilização de metodologias e práticas mais adequadas a um projeto de software é fator fundamental para se obter uma melhor qualidade nos resultados.

REFERÊNCIAS

- ALMEIDA, J. A.; RANGEL, H. J. L.; SILVA, T. A. G.; MEDEIROS, F. P. A.; LIRA, H. B. Ambiente Web com Boas Práticas para Aprendizagem de Engenharia de Software. In: JORNADA INTERNACIONAL DE PESQUISA, ENSINO, EXTENSÃO, INOVAÇÃO E SUSTENTABILIDADE DO IFPB (I JIPEEIS), 1., 2017, João Pessoa. **Anais...** João Pessoa: IFPB, 2017.
- ALMEIDA, S. L. F.; DUNNING, A. K. M. P.; FREITAS, R. M. S.; RODRIGUES, N. N.; LIMA, C. D. Q. Práticas ágeis em processos de software: análise e aplicação no GPES-IFPB. In: SEMANA DE CIÊNCIA E TECNOLOGIA DO IFPB, 12., 2017, João Pessoa. **Anais...** João Pessoa: IFPB, 2018.
- ALMEIDA, S. L. F.; MORAIS, A. D. S.; SILVA, S. C.; ALENCAR, F. P.; LIRA, H. B. Ambiente de Boas Práticas Reconfiguráveis em Engenharia de Software. In: SEMANA DE CIÊNCIA E TECNOLOGIA DO IFPB, 11., 2016, João Pessoa. **Anais...** João Pessoa: IFPB, 2017.
- AMBLER, S. W. Agile Testing strategies. In: **Amblysoft**, [2016]. Disponível em: <<http://www.amblysoft.com/essays/agileTesting.html>>. Acesso em: fev. 2017.
- BASHAM, B.; SIERRA, K.; BATES, B. **Head First Servlets & JSP**. 2nd. ed. Sebastopol, CA, EUA: O'Reilly, 2008.

BUZAN, T. **Mapas mentais**: métodos criativos para estimular o raciocínio e usar ao máximo o potencial do seu cérebro. Rio de Janeiro: Sextante, 2009.

ENDEAVOR BRASIL. **Storytelling**: separando as marcas que vêm a passeio daquelas que vêm pra ficar. [2015]. Disponível em: <<https://endeavor.org.br/storytelling/>>. Acesso em: mar. 2017.

FOX, C. **Introduction to software engineering design**: processes, principles, and patterns with UML 2. 1st ed. Boston: Pearson, 2006.

HENRIK, K. **Scrum e XP direto das Trincheiras**. 1. ed. São Paulo: C4Media, 2007.

HEUSER, C. A. **Projeto de Banco de Dados**. 4. ed. Porto Alegre: Sagra Luzzatto, 1998. (Série Livros Didáticos, n. 4).

JALOTE, P. **An Integrated Approach to Software Engineering**. 3. ed. New York: Springer, 2005.

MORAIS, A. D. S.; LIMA, C. D. Q.; RANGEL, H. J. L.; LIRA, H. B.; RODRIGUES, N. N.; ALMEIDA, S. L. F.; BARBOSA, T. C. Proposta de Framework de Gerenciamento Ágil de Projetos do Grupo de Pesquisa em Engenharia de Software do IFPB. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO (CSBC), 37.; ENCONTRO NACIONAL DE COMPUTAÇÃO DOS INSTITUTOS FEDERAIS, 4., 2017, São Paulo. **Anais...** São Paulo: Sociedade Brasileira de Computação, 2017a.

MORAIS, A. D. S.; RANGEL, H. J. L.; LIRA, H. B.; RODRIGUES, N. N.; ALMEIDA, S. L. F.; SILVA, S. C.; MEDEIROS, F. P. A. Repositório de Práticas em Gerenciamento de Projetos de Software. In: SEMANA DE CIÊNCIA E TECNOLOGIA IFPB, 11., 2016, João Pessoa. **Anais...** João Pessoa: IFPB, 2017b.

PÁDUA, S. I. D. **Investigação do processo de desenvolvimento de software a partir da modelagem organizacional, enfatizando regras do negócio**. 2001. 145 f. Dissertação (Mestrado em Engenharia de Produção) – Universidade de São Paulo, São Carlos, 2001.

PMI – Project Management Institute. **Um guia do Conhecimento em gerenciamento de projetos: Guia PMBOK®**. 5. ed. São Paulo: Saraiva, 2014.

PRESSMAN, R. S. **Engenharia de Software**. 3. ed. São Paulo: Makron Books, 1995.

PRESSMAN, R. S. **Engenharia de Software**. 6. ed. Porto Alegre: McGrawHill, 2006.

PRODANOV, C. C.; FREITAS, E. C. **Metodologia do trabalho científico**: métodos e técnicas da pesquisa e do trabalho acadêmico. 2. ed. Novo Hamburgo: Feevale, 2013. Disponível em: <<http://www.feevale.br/Comum/midias/8807f05a-14d0-4d5b-b1ad-1538f3aef538/E-book%20Metodologia%20do%20Trabalho%20Cientifico.pdf>>. Acesso em: fev. 2017.

RODRIGUES, N. N. Praticando Qualidade de Software: Ensinando e Aprendendo seus Valores através de Ambiente Real. In: SIMPÓSIO BRASILEIRO DE SISTEMAS DE INFORMAÇÃO (SBSI), 9., 2013, João Pessoa. **Anais...** Porto Alegre: Sociedade Brasileira de Computação, 2013. p. 475-486.

RODRIGUES, N. N.; ESTRELA, N. V. A. Simple Way: Ensino e Aprendizagem de Engenharia de Software Aplicada através de Ambiente e Projetos Reais. In: SIMPÓSIO BRASILEIRO DE SISTEMAS DE INFORMAÇÃO (SBSI), 8., 2012, São Paulo. **Anais...** Porto Alegre: Sociedade Brasileira de Computação, 2012. p. 722-733.

SILVA, S. C.; ALMEIDA, S. L. F.; RANGEL, H. J. L.; LIRA, H. B.; RODRIGUES, N. N.; MEDEIROS, F. P. A. Repositório de Práticas em Processos de Desenvolvimento de Software. In: SEMANA DE CIÊNCIA E TECNOLOGIA IFPB, 11., 2016, João Pessoa. **Anais...** João Pessoa: IFPB, 2017.

SOARES, S. Tudo é Software: qual é a importância da engenharia de software para o mercado e para a geração de conhecimento?. **Computação Brasil**, n. 28, p. 7-10, 2015.

SOMMERVILLE, I. **Engenharia de Software**. 8. ed. São Paulo: Addison-Wesley Brasil, 2007.