

Agile Short Unified Process – ASUP: Uma metodologia híbrida apoiada na adaptação do framework Scrum e do modelo Unified Process

Mauro Borges França¹, Alexandre Cardoso², Edgard Afonso Lamounier Junior², Camila Tavares Mota²

mauro@ifm.edu.br; alexandre@ufu.br; lamounier@ufu.br; camilatavares@ufu.br

¹ Instituto Federal do Triângulo Mineiro (IFTM), 38064-190, Uberaba-MG, Brasil

² Universidade Federal de Uberlândia (UFU), 38400-902, Uberlândia-MG, Brasil

DOI: [10.17013/risti.46.71-86](https://doi.org/10.17013/risti.46.71-86)

Resumo: Encontrar uma metodologia de desenvolvimento de software que seja aderente às características de fábricas de software é um grande desafio para pesquisadores. Neste sentido, este trabalho apresenta uma metodologia híbrida, denominada Agile Short Unified Process – ASUP. A proposta tem como base elementos extraídos do framework Scrum inseridos nas quatro fases do Unified Process – UP: Concepção, Elaboração, Construção e Transição. Em cada fase encontra-se um ciclo de vida representada e tipificada com todos os seus elementos. Para avaliar a eficiência, o ASUP foi aplicado em quatro projetos de duas instituições federais de educação profissional - IFTM e IFNMG. Após a aplicação da metodologia, alguns apontamentos foram realizados pelas equipes como forma de *feedback* e suas percepções finais foram de ganhos significativos nas atividades de gerenciamento dos projetos. Por fim, nas duas instituições, a metodologia foi aprovada e deixou de ser apenas sugestiva para ser considerada aplicável em outros projetos.

Palavras-chave: Scrum; Unified Process; Metodologias Híbridas; Atividades de Gerenciamento de Projetos.

Agile Short Unified Process – ASUP: A hybrid methodology supported by the adaptation of the Scrum framework and the Unified Process model

Abstract: A great challenge for researchers is to find out a software development methodology that can be adhered to the characteristics of software factories. In this sense, this work presents a hybrid methodology, called Agile Short Unified Process – ASUP. The proposal is based on elements extracted from the Scrum framework inserted in the four phases of the Unified Process – UP: Conception, Elaboration, Construction and Transition. We can find a life cycle represented and typified with all its elements in each stage. The ASUP was applied in four projects of two federal institutions of professional education (IFTM and IFNMG) in order to evaluate its

efficiency. After applying this methodology, some notes were made by the teams as a form of feedback, and their final perceptions realized meaningful gains in project management activities. In the end, the methodology was approved and was no longer only a suggestive one, but being considered perfectly possible to be applied in other projects, in both institutions.

Keywords: Scrum; Unified Process; Hybrid Methodologies; Project Managing Activities.

1. Introdução

Várias mudanças na forma de desenvolver projetos ocorreram durante toda história do software. Disciplinas de Engenharia de Software emergiram em consequência da “Crise de Software” na década de 1970. Esse momento motivou o surgimento das metodologias de desenvolvimento de sistemas, sendo o seu primeiro modelo denominado cascata (Guéhéneuc & Khomh, 2019).

Outro fator que agitou mudanças na área de Engenharia de Software aconteceu no ano de 2001, quando dezessete profissionais especializados em metodologias consideradas leves estabeleceram princípios e características comuns em cada uma, resultando na “Aliança Ágil”, a qual constituiu o documento denominado “Manifesto Ágil” (Mancl & Fraser, 2019).

Estes dois momentos históricos, a “Crise de Software” e o “Manifesto Ágil”, foram marcantes em suas respectivas épocas, pois incentivaram as empresas a aprimorarem suas maneiras de trabalhar em projetos de software (Stol & Fitzgerald, 2018).

Para Barbosa et al (2021), a aplicação certa de métodos e técnicas em gerenciamento de projetos mitiga as incertezas epistêmicas em projetos de software e pode representar um diferencial competitivo para a indústria de software.

Para que a condução dos projetos de software seja aprimorada nas organizações é de suma importância o uso de uma metodologia. Dentre elas destacam-se os modelos prescritivos ou tradicionais e as mais atuais sendo denominadas ágeis. Algumas empresas julgam que o software que produzem pode ser compreendido simplesmente ao ler seu código-fonte - modelos ágeis - e outras documentam seus produtos de forma intensiva - modelos tradicionais (Gonçalves, 2018).

Para Castilla (2014), alguns acadêmicos e profissionais propuseram diferentes abordagens com o objetivo de encontrar métodos de desenvolvimento apropriados para características de projetos de softwares específicos. Neste sentido, os estudos apontam para o uso de metodologias híbridas que combinam duas ou mais metodologias de desenvolvimento de software com a finalidade de alavancar os respectivos pontos positivos e melhorando, assim, todo o processo de desenvolvimento.

Para Kuhrmann et al (2017), uma abordagem híbrida é qualquer combinação de metodologias ágeis e tradicionais, a qual uma unidade organizacional adota e personaliza às suas próprias necessidades de contexto (por exemplo, domínio de aplicação, cultura, processos, projeto, estrutura organizacional, técnicas, tecnologias, etc.).

Considerando a possibilidade de contribuição, este trabalho apresenta o Agile Short Unified Process - ASUP, uma metodologia baseada nas fases do Processo Unificado - PU

- com elementos extraídos do *framework* Scrum. Partindo desse panorama, levanta-se a seguinte hipótese: “A combinação de estratégias presentes nos modelos ágeis (Scrum) com elementos dos Processos Unificados - PU - irá prover melhorias na dinâmica, condução e gestão de projetos conduzidos em iniciativas oriundas de instituições de ensino público”.

Portanto, como objetivo geral, esse trabalho propõe um conjunto de técnicas computacionais e gerenciais, adequando a integração de aspectos relevantes do Processo Unificado e do *framework* Scrum, a serem aplicados em projetos de médio e pequeno porte motivados em fábricas de software localizadas em instituições públicas de ensino.

Como resultado da avaliação de produtividade da proposta, o trabalho foi aplicado nos setores de Tecnologia da Informação e Comunicação das seguintes instituições:

- Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro localizado na cidade de Uberaba (MG);
- Instituto Federal de Educação, Ciência e Tecnologia do Norte de Minas Gerais localizado na cidade de Montes Claros (MG);

2. Fundamentação Teórica

Para entender a proposta deste trabalho, foi elaborado um mapa conceitual com os elementos envolvidos na pesquisa. De acordo com Serrat (2017), esta técnica permite representar os componentes da temática do trabalho investigado e organizar os conceitos, temas ou tarefas em um tópico central. Nessa linha de raciocínio, a metodologia ASUP, sendo o foco central da pesquisa, apresenta uma relação das combinações de práticas utilizadas no *framework* Scrum, relacionando-as às fases do Processo Unificado. Além de apresentar as características dos modelos envolvidos na pesquisa, outros elementos que se relacionam com o tema são demonstrados no mapa conceitual para simbolizar suas relações.

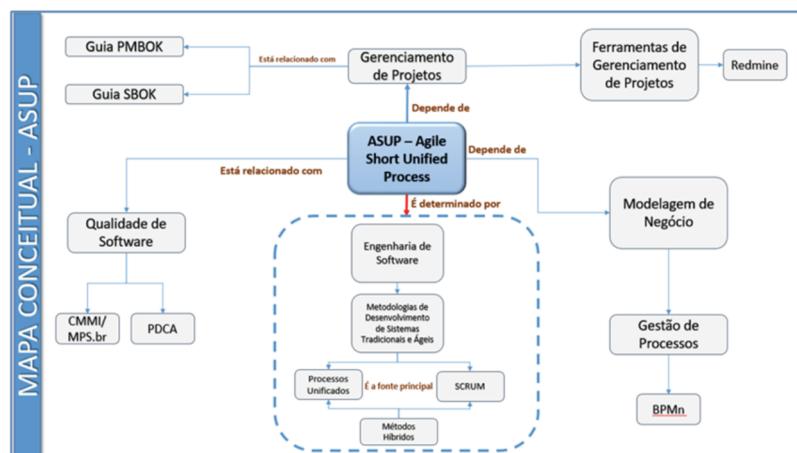


Figura 1 – Mapa Conceitual

No mapa destaca-se o ASUP como ponto central, sendo delimitado pelo pontilhado com a mesma cor da caixa do ASUP, demonstrando que se trata da principal área da investigação deste trabalho. A seta com a descrição “É determinado por” deixa claro que o ASUP está fortemente relacionado com a área de “Engenharia de Software” juntamente com suas extensões “Metodologia de Desenvolvimento de Sistemas”, “Processo Unificado”, “Scrum” e “Métodos Híbridos”.

O estudo investigado também depende de outros elementos, como o Gerenciamento de Projetos, pois utiliza diretrizes conhecidas na 6^a edição do Guia de referência do PMBOK. Logo, um dos pontos chaves é o uso de uma ferramenta de gerenciamento de projetos *open source* denominada *Redmine*, permitindo a estruturação e a acomodação do rol de tarefas para relacionar com o uso da metodologia.

Diante da estrutura base da pesquisa demonstrado no mapa conceitual, serão apresentados apenas os conceitos básicos das duas metodologias aderentes à pesquisa e o contexto de metodologias híbridas para o entendimento da proposta.

2.1. Framework Scrum

De acordo com Morandini et al (2021), Scrum é considerado atualmente uma das metodologias mais utilizadas em projetos de software, surgiu em meados de 1990 e foi idealizado por Ken Schwaber e Jeff Sutherland que se apoiou nos métodos Lean (ciclo do processo de produção da Toyota) e OODA (ciclo da aviação de combate dos USA). Portanto, da junção dessas duas metodologias nasceu o Scrum, que é definido como um *framework* para desenvolver, entregar e manter produtos complexos. Essa definição consiste em papéis, eventos, artefatos e regras e os mantém integrados. Para seus idealizadores, o Scrum é um método leve, simples de entender, porém de difícil domínio (Sutherland, 2016).

Como no Processo Unificado, o Scrum também emprega uma abordagem iterativa e incremental para aperfeiçoar a previsibilidade e o controle de riscos. Além disso, ele tem como pilares a transparência, a inspeção e a adaptação.

O Scrum é composto pelo Time Scrum que desenvolve suas atividades conforme suas cerimônias e dentro dos *times box* definidos. Como papéis o Scrum é dividido em três tipos que compõe o time: Product Owner, Scrum Master e o Time de Desenvolvimento (Shafiee et al, 2020).

A Sprint é considerada o coração do Scrum e possui um *time-boxed* de um mês ou menos, durante o qual um incremento de produto potencialmente liberável é criado. A Sprint é definida por meio da reunião de Planejamento da Sprint e esse é elaborado de forma colaborativa entre todos do time Scrum. Como forma de acompanhar as etapas do projeto, a cerimônia da reunião diária é utilizada pelo Time de Desenvolvimento num evento *time-boxed* de 15 minutos para que o time possa sincronizar as atividades e criar um plano para as próximas vinte e quatro horas. No final de cada Sprint é realizada a reunião de sua revisão que tem como objetivo inspecionar o incremento e adaptar o Product Backlog, caso necessário. Por fim, a Retrospectiva da Sprint é o momento oportuno para que o Time Scrum inspecione como a Sprint foi conduzida, identificando e ordenando os pontos positivos e outros que devem ser aprimorados.

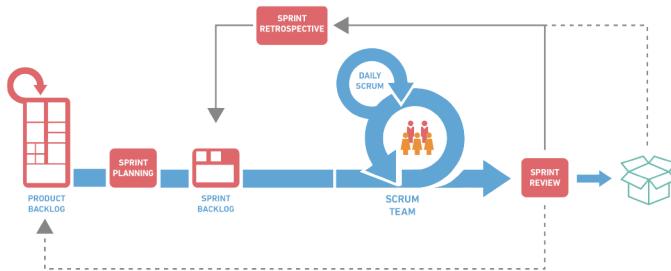


Figura 2 – Ciclo de Vida do Processo Scrum

O resumo do Scrum permite uma análise das várias cerimônias e estruturas que são consideradas próximas às características do trabalho proposto.

2.2. Unified Process

O Processo Unificado tem como pilar ser dirigido por Casos de Uso, centrado em arquitetura e ser iterativo e incremental. Cada ciclo termina com a liberação de uma versão do sistema para os clientes. No Processo Unificado, cada ciclo contém quatro fases (Wazlawick, 2019). Uma fase é simplesmente o tempo decorrido entre dois marcos principais em que gerentes tomam decisões importantes sobre se prosseguem ou não com o desenvolvimento e, se a resposta for afirmativa, eles deliberam sobre o que é necessário fazer em relação ao escopo, ao orçamento e ao cronograma do projeto (Shafiee, 2020). A Figura 3 apresenta as fases e os principais marcos do Processo Unificado.

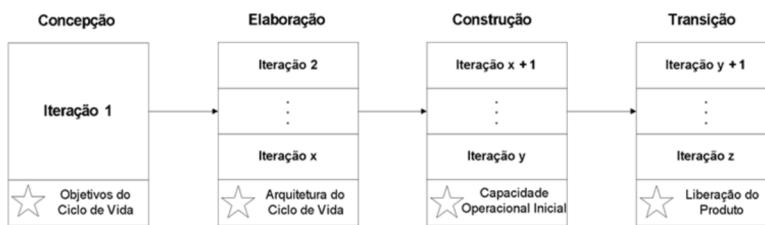


Figura 3 – Fases e Marcos Principais do Processo Unificado

A seguir são apresentadas visões gerais de cada fase do Processo:

- **Concepção:** O objetivo principal da fase de Concepção é estabelecer a viabilidade do sistema proposto. A definição do escopo, a arquitetura candidata e a análise econômica são os aspectos predominantes desta fase.
- **Elaboração:** Esta fase tem como principais aspectos a captura dos requisitos funcionais válidos restantes, o estabelecimento de uma base arquitetônica, a abordagem dos possíveis riscos significativos e a preparação do plano para orientar a próxima fase (construção).

- Construção: Nesta fase fica a responsabilidade da construção do sistema em questão. Esse deverá ser capaz de operar bem em ambientes de teste, no qual o próprio cliente os realiza.
- Transição: O principal objetivo da fase de Transição é entregar o sistema completamente funcional aos clientes. O principal marco associado à fase de Transição é chamado de Liberação do Produto.

O ASUP traz em seu ciclo de vida as quatro fases representadas no Processo Unificado, sendo este o principal motivo de sua familiaridade.

2.3. Metodologias Híbridas

A crescente demanda por projetos de software em diferentes domínios motivam oportunidades para que a indústria de software aprimore a forma de desenvolver seus projetos. Em função disso, atualmente há muitos estudos que envolvem as metodologias tradicionais e ágeis (Muñoz et al, 2019). Para Mirza & Datta (2019), tais métodos apresentam pontos fortes e fracos, sendo que as metodologias ágeis superam algumas das fraquezas apresentadas nas tradicionais.

Segundo Dingsøyr et al (2018), os métodos tradicionais são aplicados em sistemas de informações que demandam um planejamento rígido e extenso, confrontando as características adotadas pelos métodos ágeis, nos quais há maior flexibilidade na construção dos projetos de software, pois aplicam técnicas por meio de aprimoramento e testes contínuos com base em feedback e modificações ágeis.

Embora nos últimos anos as metodologias ágeis tenham sido usadas por empresas de desenvolvimento de software, ainda existe uma alta taxa de falhas de software quando comparadas com os principais processos de engenharia (Dhir et al, 2019).

Diante de todos esses julgamentos, a adaptação de metodologias se tornou uma prática comum na adesão ao processo de produção de software. Dessa forma, as indústrias de software buscam customizar os processos de software em sintonia com sua estrutura organizacional. Para isso este trabalho apresenta o ASUP para ser discutido no meio científico.

3. Metodologia Proposta

3.1. Visão geral do ASUP

Após entender as características dos dois principais modelos e compreender que as metodologias híbridas podem aprimorar no processo de soluções em projetos de software, este trabalho apresenta uma metodologia híbrida com base em elementos extraídos do Scrum e do Processo Unificado, a fim de avaliar seu potencial em projetos oriundos de setores de tecnologia da informação e comunicação em duas intituições de ensino profissional.

Do Processo Unificado as fases foram representadas, pois acredita-se que facilita aos envolvidos no projeto a identificação do estágio em que se encontram as atividades de uma determinada Sprint. Já no Scrum a representação do ciclo com o uso das Sprints

foram consideradas, pois representa as entregas no progresso iterativo e incremental. Ainda com aproximação do Scrum, o ASUP direciona as inspeções, com suas cerimônias de *time-box*. As inspeções permitem um direcionamento das atividades realizadas, tanto de cunho técnico como de acompanhamento das entregas. A figura 4 apresenta o ciclo de vida do modelo ASUP com suas cerimônias, artefatos e papéis envolvidos.

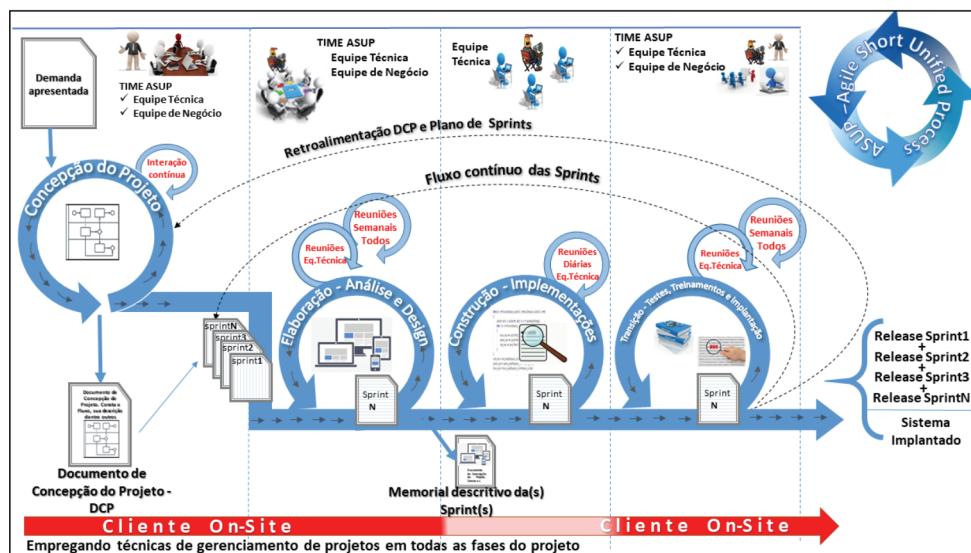


Figura 4 – Ciclo de vida do processo ASUP

Cada fase irá produzir um conjunto de atividades e, consequentemente, um conjunto de artefatos para documentar as etapas do processo de desenvolvimento. Logo, ao final de cada fase, espera-se obter tais artefatos, sejam eles diagramáticos ou textuais, a depender da fase em questão.

Para um melhor entendimento da metodologia demonstrada, faz-se necessário apresentar os elementos nela contida, dividindo-os em quatro pilares de sustentação:

- Atores envolvidos;
- Ciclos de eventos fragmentados em fases;
- Envolvimento bem definido do cliente no projeto;
- Uso de ferramenta de gerenciamento de projetos para acomodar as tarefas e artefatos.

3.2. Atores da metodologia ASUP

Normalmente os projetos possuem dois segmentos de times, um composto por papéis de quem está desenvolvendo o serviço técnico e o outro pelos componentes que estão demandando o serviço, também conhecidos como clientes. Para interações das atividades trabalhadas entre esses dois segmentos, a metodologia ASUP propõe uma divisão em

dois tipos de equipes. O objetivo dessa divisão é estabelecer uma comunicação efetiva na condução das etapas do projeto, descritos a seguir.

Equipe da Regra de Negócio ou, simplesmente, Equipe de Negócio

Esta equipe tem como responsabilidade repassar todo conhecimento da regra de negócio do projeto, bem como eliciar todos os requisitos a serem estabelecidos. A estreita comunicação com a equipe técnica permitirá transformar os problemas apresentados em soluções de software. Para isso, o ASUP propõe duas divisões:

- Patrocinador(es);
- Stakeholder(s).

Equipe Técnica (equipe de profissionais técnicos)

A equipe técnica é a responsável por conduzir as atividades de cunho técnico que envolve o projeto. Cada papel deverá ter um rol de responsabilidades, a fim de organizar e estabelecer uma comunicação que permita alcançar as metas e os objetivos desejados. Deste modo o ASUP propõe três papéis:

1. Gerente de Projetos;
2. Coordenador ou Líder de Equipes;
3. Desenvolvedores.

3.3. Ciclos fragmentados em fases

Concepção Inicial do Projeto

O início de qualquer projeto parte de uma pessoa ou grupo de pessoas que possui o interesse em desenvolver um produto ou serviço para solucionar um problema ou aprimorar um processo qualquer. Portanto, ao ser apresentada uma demanda e essa aprovada para ser desenvolvida no projeto, o ASUP entra no primeiro ciclo denominado “Concepção do Projeto”. Poderá acontecer do Patrocinador e do Gerente de Projetos não entrarem em acordo, seja por questões de viabilidade financeira, seja por outro fator que justifique a não continuidade do projeto e, sendo assim, ele será encerrado. Porém o método proposto neste trabalho tem como foco apresentar o processo somente após o aceite da demanda por parte do Patrocinador e do Gerente de Projetos.

Para a compreensão final da importância desta fase e como destaque desse ciclo, a figura 4 apresentada na seção da visão geral do ASUP, demonstra um degrau no desenho de seu ciclo. Esse degrau representa que esse primeiro ciclo será realmente a base da estrutura da metodologia. Isso significa que mesmo podendo ser retroalimentado o planejamento das Sprints e todo o processo de desenvolvimento de cada ciclo, todos os procedimentos serão orientados com base nas atividades desenvolvidas nessa fase.

Resumo dos eventos/tarefas a serem desenvolvidos nesta fase:

1. Reunião de instalação dos trabalhos para a compreensão do contexto geral do problema apresentado com algum artefato que motivou a demanda;
2. Interações contínuas (reuniões formais ou informais) para lapidar o fluxo do processo que envolve o contexto do domínio da aplicação;

3. Elaboração e validação do fluxo Business Process Model and Notation -BPMn - de maneira colaborativa entre equipe técnica e equipe de negócio;
4. Elaboração e validação do planejamento das Sprints a serem trabalhadas na construção do projeto;
5. Elaboração e validação do Documento de Concepção do Projeto;

Resumo dos artefatos da fase:

1. Documento que apresentou a demanda (e-mail, ofício, história inicial apresentada ou outro artefato que represente o *start* do projeto);
2. Fluxo BPMn (importante que registre todas as versões do documento para demonstrar a evolução do entendimento do processo);
3. Planilha com o planejamento das Sprints a serem trabalhadas no projeto (planilha, tabela, website colaborativo ou outro documento que represente a divisão das Sprints, bem como suas estimativas e prioridades);
4. Documento de Concepção do Projeto (documento com mais detalhes e informações dos demais artefatos citados acima).

Uma observação importante é que todos os artefatos são dinâmicos e podem a qualquer momento serem retroalimentados conforme identificadas alterações no projeto e desde que seja acordado entre as equipes técnica e de negócio.

Elaboração – Análise e Design

Dante do Planejamento das Sprints, o ciclo denominado “Elaboração – Análise e Design” tem como objetivo principal o detalhamento dos requisitos de um grupo de funcionalidades que compõe uma Sprint. A equipe técnica em conjunto com a equipe de negócio tem o desafio de transformar as informações trabalhadas nas interações semanais em protótipos a serem lapidados até que possam ser devidamente utilizadas no contexto incremental do projeto. Essa fase também tem característica iterativa e incremental, pois por meio de várias interações as equipes de negócio e técnica estarão juntas buscando o objetivo em comum. Nesta fase a equipe técnica elabora um artefato denominado “memorial descritivo”, o qual estabelece todos os elementos envolvidos e aprovados pela equipe de negócio.

Um evento que se aproxima das atividades desenvolvidas no *framework* Scrum são as reuniões diárias. O ASUP sugere, mas não impõe que esse evento aconteça, pois podem existir características de projetos que inviabilizam tal encontro. O evento obrigatório imposto pelo ASUP são as reuniões semanais, elas sim permitirão que o time ASUP acompanhe se as metas estabelecidas foram cumpridas e também permitirão planejar as próximas metas. É importante que a reunião aconteça para avaliar o andamento dos trabalhos da Sprint. Para que esse processo seja transparente, todas as tarefas deverão estar devidamente registradas no ambiente de gerenciamento de projetos.

A responsabilidade de avaliar e analisar se as tarefas planejadas estão sendo cumpridas é do líder da equipe de desenvolvedores. Ele deverá, por meio do ambiente de gerenciamento de projetos, promover tarefas e acompanhar se elas estão sendo realizadas ou não. Os desenvolvedores deverão realizar o lançamento de atividades vinculadas às tarefas para permitir ao líder verificar e acompanhar sua progressão.

Com o artefato denominado memorial descritivo aprovado pelo líder da equipe juntamente com os desenvolvedores, o ciclo é finalizado e as atividades são encaminhadas para a próxima fase.

Ressalta-se que o envolvimento da equipe de negócio nesses dois primeiros ciclos é constante e todo o processo somente é continuado com a anuência e definição das duas equipes (técnica e de negócios). Isso é bem representado na figura 4 do ciclo ASUP por meio da tarja vermelha, a qual é constante nos dois ciclos conforme demonstrado.

Resumo dos eventos/tarefas/artefatos a serem desenvolvidos nesta fase:

1. Reuniões semanais para identificar, detalhar e validar os requisitos;
2. Elaboração dos Protótipos de Interfaces e dos Fluxos BPMn quando necessário;
3. Arquitetar e estruturar banco de dados;
4. Elaboração do Memorial descritivo.

Resumo dos artefatos produzidos na fase de concepção:

1. Protótipo de Interfaces por Sprints;
2. Fluxo BPMn (quando necessário, a depender da complexidade da Sprint)
3. Memorial descritivo.

Construção – Implementações

O ciclo “Construção – Implementações” apresenta a característica mais técnica dentro do modelo, pois representa a codificação e transformação dos artefatos documentais em códigos para serem compilados em um projeto de software. Este trabalho tem como base as definições desenvolvidas no memorial descritivo apresentado.

Uma observação importante dessa fase é a barra de presença do cliente denominada “*Client on site*”. A figura 4 apresentada na visão geral do ASUP, demonstra uma coloração bem mais clara na barra, pois no processo de codificação, a equipe de negócio poderá até ser envolvida, porém com menor intensidade do que nos demais ciclos. Isso ocorre apenas quando a equipe técnica encontra alguma necessidade de *feedback*, podendo o processo retornar para o ciclo anterior para averiguar o contexto do problema apresentado. Esse retorno poderá reestabelecer o *workflow* do processo ASUP, mas deve ser acordado por ambas equipes.

Resumo dos eventos/tarefas/artefato a serem desenvolvidos nesta fase:

1. Código fonte;
2. Teste de unidade da Sprint.

Transição – Testes, Treinamentos e Implantação

O último ciclo da Sprint é o definido como “Transição – Testes, Treinamentos e Implantação”. Ele é responsável por envolver a equipe de negócio do projeto no processo de teste dos componentes da Sprint. Caso os testes realizados pelos usuários sejam aceitos, eles passarão pelo processo de treinamento quando houver mais de um usuário a utilizar o sistema a ser liberado.

Se ao findar uma determinada Sprint, o time ASUP detectar necessidade de retroalimentação do planejamento das Sprints por motivos acordados entre ambas as partes, isso será realizado. Caso não haja ocorrências nesse sentido, segue-se o fluxo contínuo para a próxima Sprint, iniciando-se o processo novamente no ciclo de Elaboração – Análise e Design da Sprint seguinte.

Resumo dos eventos/tarefas a serem desenvolvidos nessa fase:

1. Reuniões para validação da Sprint;
2. Reunião para avaliar retroalimentação do Plano de Sprints;
3. Realização de testes com as equipes juntas.

Resumo dos artefatos produzidos na fase de concepção:

1. Relatório de testes;
2. Manuais da Sprint, quando necessário, para facilitar o entendimento das funcionalidades que a compõe;
3. Aceite final com anuência para implantar a Sprint no ambiente de produção.

3.4. Artefatos

A metodologia ASUP não restringe que se utilize apenas os artefatos especificados no desenho de seu ciclo e descritos em suas fases. Isso porque, dependendo do tipo do projeto, ele deverá ter outros diagramas e documentos que fortaleçam sua composição e organização.

A sugestão dos artefatos contidos no ASUP procura atender as necessidades de comunicação e transparência do processo. Os artefatos básicos gerados na metodologia ASUP estão representados em suas referidas fases conforme já mencionado.

Os artefatos recomendados deverão estar devidamente acomodados e registrados no ambiente de gerenciamento de projetos *Redmine*. O único artefato que não constará no ambiente será o código fonte do projeto que fará parte da estrutura armazenada em repositório apropriado e definida pela equipe técnica.

3.5. O envolvimento do cliente do projeto

Conforme já mencionado nos ciclos da metodologia ASUP, a presença da equipe de negócios do projeto é representada no desenho do ciclo de vida do processo por meio de uma barra de cor vermelha com o nome “*Client On-Site*”. Observa-se a intensidade da cor, pois representa o nível de envolvimento do cliente no projeto. O termo “*Client on-site*” significa que a equipe de negócios estará presente de forma atuante em todo o momento em que a barra esteja na cor vermelha intensa. Analisando a barra de envolvimento do cliente, é bem nítido que ele está fortemente envolvido nos ciclos de concepção, elaboração e transição e com menor participação no ciclo de construção. Isso se deve ao motivo de ser um ciclo com particularidades específicas do papel de desenvolvedor. Esse transformará em códigos os artefatos que foram devidamente construídos com envolvimento do cliente de forma intensa e, consequentemente, tais códigos se tornarão componentes de software da Sprint.

3.6. Ferramenta web de gerenciamento de projetos com aderência à metodologia ASUP

A metodologia ASUP é apoiada por meio de uma ferramenta web de gerenciamento de projetos denominada *Redmine*. Essa ferramenta é uma solução *open source* e apresenta uma série de funcionalidades para facilitar a condução de projetos. *Redmine* foi desenvolvido com o framework denominado *Ruby on Rails*, e lançado sob os termos GNU - General Public License (Kaminsk, 2019).

Abramova et al (2016), realizou uma pesquisa comparando as principais ferramentas de gerenciamento de projetos de código aberto e soluções proprietárias. Nessa pesquisa, o autor apresentou que das soluções livres pesquisadas, Redmine tem seu destaque significativo por possuir várias funcionalidades equiparando-se às soluções proprietárias.

Arias (2019) cita em seu trabalho que o Redmine “*serviu de plataforma para a comunicação, avaliação do trabalho dos membros da equipe, retroalimentação, consultas de documentos e socialização dos resultados obtidos em cada etapa*”.

Pelos motivos apresentados, o Redmine foi a escolhida para ser utilizada e recomendada para aplicar a metodologia ASUP.

Nas duas instituições do caso de uso o *Redmine* foi implantado em servidores próprios e devidamente acomodados em seus Data Centers. Portanto, existem outros ensaios em andamento sendo realizados e atualmente o ASUP possui um domínio com estrutura própria (<http://asup.net.br>).

4. Aplicabilidade da Metodologia Proposta

Para avaliar a eficiência e eficácia do ASUP, a metodologia foi aplicada em duas instituições de educação profissional nos seus setores de desenvolvimento. Foram dois projetos vinculados à Diretoria de Tecnologia da Informação e Comunicação do Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro – IFTM e dois projetos vinculados à Diretoria de Tecnologia da Informação e Comunicação do Instituto Federal de Educação, Ciência e Tecnologia do Norte de Minas – IFNMG.

Nº	Descrição dos projetos	Local
1	Projeto Datas Comemorativas – DC	DTIC-IFTM
2	Projeto Quadro Informativo – QI	DTIC-IFTM
3	Projeto Sistema de Graduação UAB/IFNMG	DTIC-IFNMG
4	Projeto Sistema de Pós-Graduação UAB/IFNMG	DTIC-IFNMG

Tabela 1 – Projetos que utilizaram o ASUP

Anterior à utilização do ASUP, em ambas as instituições eram utilizadas metodologias diferentes da proposta deste trabalho, sendo o IFTM uma adaptação do Processo Unificado e o IFNMG uma adaptação do Scrum.

Para as duas instituições, a aplicação da metodologia ASUP teve início com um processo de treinamento junto ao time escolhido para utilizá-la. Após o treinamento e o conhecimento prévio da demanda apresentada por meio de um documento interno das fábricas, criou-se os projetos no ambiente Redmine, definindo os membros do time ASUP e os escopos iniciais dos projetos conforme suas definições.

Com as equipes treinadas, os projetos foram conduzidos e em vários momentos foram sendo ajustados os entendimentos quanto aos procedimentos utilizados no processo ASUP. Todas as atividades dos projetos foram conduzidas de forma orientada e após a aplicação da metodologia com seus ritos e cerimônias, realizou-se uma reunião com os times para levantar características a serem aperfeiçoadas. Diante disso, as equipes técnicas que utilizou o ASUP listaram pontos importantes apresentados abaixo:

1. Como o time era composto por 1 Coordenador e 3 desenvolvedores, em alguns momentos, por questões de férias e afastamentos, o projeto trouxe problemas de sequenciamento. Problemas esses que podem ser classificados apenas como gerenciamento de time;
2. O time tinha integrantes que não conheciam algumas tecnologias utilizadas no projeto e, por isso, foi necessário realizar um nivelamento entre os integrantes;
3. Como os projetos demandavam integrações com outros módulos, foi detectado a necessidade de envolver um profissional para trabalhar com a integração do Banco de Dados quando houvesse a presença de um Administrador de Banco de Dados - DBA na equipe, por se tratar de um projeto de software escalável.
4. As equipes deveriam ter cuidado com a padronização de cada módulo construído e evitar que no resultado final, na organização da tela, na documentação manual e no tutorial o usuário não desenvolvesse critérios diferentes em cada módulo.
5. De alguma forma, o gerente do projeto deveria “fiscalizar” o alinhamento na produção dos módulos para que permaneçam integrados no Sistema de Gestão Integrado - ERP;
6. Por outro lado, a autonomia das equipes nos permite uma diferenciação nas tarefas individuais, quebrando o monotonismo de repetição constante - só programar ou só testar. Ainda traz um fator crucial que é a entrega de partes do projeto ao cliente, tornando-o mais próximo das etapas do desenvolvimento, fator esse não existente no método utilizado anteriormente.
7. Como auxílio para o entendimento da metodologia, talvez poderia ser feito um tutorial aplicado a um caso hipotético bem simples, demonstrando o uso dos elementos gerados no processo de cada etapa e o andamento do começo, meio e fim do método. Isso ficaria interessante e prático e já ajudaria na introdução mais rápida do ASUP.

Em resumo, após analisar os apontamentos realizados pelas equipes, percebe-se três pontos distintos, a saber 1) aspectos gerenciais e estruturais e que demandam uma gestão de pessoal, 2) elaboração de manuais do uso da metodologia e da ferramenta para auxiliar sua implementação e 3) definições de padrões de projeto, aspecto primordial para evitar possíveis problemas de qualidade.

5. Conclusões

A adoção da metodologia obteve resultados significativos, aprimorando as práticas na gestão de projetos de software, e acredita-se que proporcionou melhorias 1) nos aspectos de comunicação entre as equipes por meio das interações contínuas, 2) na organização dos artefatos para uma possível rastreabilidade, 3) na dinâmica dos processos de planejamento, execução e monitoramento das atividades vinculadas aos projetos e 4) na adoção de uma ferramenta que permitiu a aplicação da metodologia. Por fim, as etapas acima propiciaram transparência nas ações desenvolvidas no projeto.

Neste sentido, a aplicação da metodologia ASUP demonstrou transformações importantes nos projetos que a utilizaram e, consequentemente, deixou um legado quanto à forma de condução de projetos de software nas instituições envolvidas. De uma forma geral, sua aplicação proporcionou melhores resultados no planejamento e execução dos projetos conduzidos e nos locais de aplicação a proposta deixou de ser sugestiva e se transformou em realidade.

6. Trabalhos Futuros

Diante dos aspectos demonstrados nessa pesquisa, algumas demandas foram apresentadas com interesse do uso do ASUP, porém projetos com características distintas, em destaque três vertentes que poderiam ser utilizadas com ASUP:

- Aplicação do ASUP em projetos de Pesquisa e Desenvolvimento P&D;
- Adequação das fases do ASUP para projetos menores, tais como iniciação científica, TCCs, monografias e teses;
- Adequação do ASUP para projetos utilizados em sala de aula, juntamente com as metodologias ativas.

Por fim, este trabalho buscou apresentar ao mundo científico as vantagens de integrar metodologias tradicionais e ágeis para alcançar melhores resultados no desenvolvimento de software conduzidos em instituições de ensino. Para isso, historicamente os métodos sempre surgiram de outros métodos e assim foram evoluindo, existindo o Scrum, o RUP e atualmente uma proposta diferente, o ASUP.

Referências

- Abramova, V., Pires, F., & Bernardino, J. (2016). Open Source vs Proprietary Project Management Tools. In: Rocha, Á., Correia, A., Adeli, H., Reis, L., Mendonça Teixeira, M. (eds) New Advances in Information Systems and Technologies. Advances in Intelligent Systems and Computing, vol 444. Springer, Cham. https://doi.org/10.1007/978-3-319-31232-3_31
- Arias O. E., Ferrá, M. De los A. L., & Curbelo, F. H. (2019). Estrategia para la evaluación orientada a la formación de la competencia trabajo virtual en equipo en docentes universitarios. Libro de Actas IN-RED 2019. 5º Congreso de Innovación Educativa y Docencia en Red, Editorial Universitat Politècnica de València, <https://doi.org/10.4995/INRED2019.2019.10346>

- Barbosa F. J., Marinho M. L. M., & Moura, P. H. (2021). Em Direção a um Modelo para Quantificação da Incerteza Epistêmica em Projetos de Software: uma Pesquisa-Ação. *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação*, (E44), 67-83, <https://doi.org/10.17013/risti.44.67-83>
- Castilla, D. (2014). *A Hybrid Approach Using RUP and Scrum as a Software Development Strategy*. UNF Theses and Dissertations. 514. <http://digitalcommons.unf.edu/etd/514>. 2014.
- Dhir, S., Kumar, D., & Singh, V.B. (2019). Success and Failure Factors that Impact on Project Implementation Using Agile Software Development Methodology. In: Hoda, M., Chauhan, N., Quadri, S., & Srivastava, P. (eds) Software Engineering. Advances in Intelligent Systems and Computing, vol 731. Springer, Singapore. https://doi.org/10.1007/978-981-10-8848-3_62
- Dingsøyr, T., Dybå, T., Gjertsen, M., Jacobsen, A. O., Mathisen, T. E., Nordfjord, J. O., Røe, K., & Strand, K. P. (2018). *Key lessons from tailoring agile methods for large-scale software development*. arXiv. <https://doi.org/10.48550/arXiv.1802.05118>
- Guéhéneuc, YG. & Khomh, F. (2019). Empirical Software Engineering. In: Cha, S., Taylor, R. & Kang, K. (eds) Handbook of Software Engineering. Springer, Cham. https://doi.org/10.1007/978-3-030-00262-6_7.
- Gonçalves, V., & Campos, C. (2018). *HCMBOK*: The Human Change Management Body of Knowledge. (3nd. ed.). CGC Press.
- Kaminski, P. (2019). *Redmine*: Gerenciamento flexível de projetos. Casa do Código
- Kuhrmann, M., Diebold, P., Münch, J., Tell, P., Garousi, V., Felderer, M., Trekttere , K., McCaffery, F., Linssen, O., Hanser, E., & Prause, C. R. (2017). Hybrid Software and System Development in Practice: Waterfall, Scrum, and Beyond. *ICSSP'17*, Paris, France, ACM. <https://doi.org/10.1145/3084100.3084104>
- Mancl, D., & Fraser, S.D. (2019). XP 2019 Panel: Agile Manifesto – Impacts on Culture, Education, and Software Practices. In: Hoda, R. (ed) Agile Processes in Software Engineering and Extreme Programming – Workshops. XP 2019. Lecture Notes in Business Information Processing, vol 364. Springer, Cham. https://doi.org/10.1007/978-3-030-30126-2_17
- Morandini, M., Coletti, T.A., Oliveira, E., & Corrêa, P.L.P. Considerations about the efficiency and sufficiency of the utilization of the scrum methodology: A survey for analyzing results for development teams *Comp. Sci. Rev.*, 39, e100314. <https://doi.org/10.1016/j.cosrev.2020.100314>
- Mirza, M. S., & Datta, S. (2019). Strengths and Weakness of Traditional and Agile Processes - A Systematic Review. *Journal of Software*, 14(5), 209–219, <https://doi.org/10.17706/jsw.14.5.209-219>

- Muñoz, M., Peralta, M., & Laporte, Y.C. (2019). Análisis de las debilidades que presentan las Entidades Muy Pequeñas al implementar el estándar ISO/IEC 29110: Una comparativa entre estado del arte y el estado de la práctica. *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação*, (E34), 85-96. <https://doi.org/10.17013/risti.34.85-96>
- Serrat, O. (2019). *Knowledge Solutions*. <https://doi.org/10.1007/978-981-10-0983-9>.
- Shafiee, S., Wautlelet, Y., Hvam, L., Sandrin, E., & Forza, C. (2020). Scrum versus Rational Unified Process in facing the main challenges of product configuration systems development. *J Syst Softw.* 170, 11073. <https://doi.org/10.1016/j.jss.2020.110732>
- Sutherland, J. (2016). *Scrum : A Arte De Fazer O Dobro Do Trabalho Na Metade Do Tempo* (2a ed.). Leya.
- Stol, K. J., & Fitzgerald, B. (2018). The ABC of software engineering research. *ACM Trans Softw Eng Methodol*, 27(3), 1-51. <https://doi.org/10.1145/3241743>
- Wazlawick, S. (2019). *Engenharia de software: conceitos e práticas*. GEN LTC.