

INTEGRAÇÃO COLABORATIVA E AGILIDADE EM PROJETOS DE TECNOLOGIA DA INFORMAÇÃO DO SENAC – SP

COLLABORATIVE INTEGRATION AND AGILITY IN INFORMATION
TECHNOLOGY PROJECTS FROM SENAC – SP

Recebido em: 31.8.2023
Aprovado em: 18.9.2023

Marcelo Diniz Augusto

*Mestre em Administração do Desenvolvimento de Negócios pela Universidade Presbiteriana Mackenzie (UPM). Master of Business Administration (MBA) em Administração de Empresas pela Fundação Armando Alvares Penteado (Faap).
Bacharel em Ciência da Computação pela Universidade Paulista (Unip).
E-mail: marcelo.daugusto@sp.senac.br*

Alberto de Medeiros Jr.

*Doutor em Engenharia de Produção pela Escola Politécnica da Universidade de São Paulo (USP).
Docente no Programa de Pós-Graduação em Administração do Desenvolvimento de Negócios da UPM.
E-mail: alberto.medeiros@mackenzie.br*

RESUMO

Este relato tecnológico analisa as metodologias de gestão de projetos de tecnologia da informação (TI) e integração das equipes no ambiente corporativo do Serviço Nacional de Aprendizagem do Comércio de São Paulo (Senac – SP). O estudo amparou-se em diferentes metodologias de gestão de projetos, demonstrando as suas utilidades e aplicabilidade e as fases que são percorridas até o momento da entrega do resultado. A adoção de um modelo gerencial teve por base um forte alinhamento aos objetivos estratégicos do negócio e às respectivas áreas para se obter a melhor solução não obstante as lacunas que possam ser identificadas durante a sua adoção. Observou-se na entidade um relativo aumento nos atrasos de entregas de projetos de tecnologia e na colaboração entre especialistas de tecnologia. O Senac – SP tem grande quantidade de projetos em fila para execução e os diferentes ativos e plataformas tecnológicas demonstram dificuldades quanto às operações, ao desenvolvimento das plataformas e à qualidade e ao suporte, o que pode afetar diretamente a entrega ou sustentação de um ativo tecnológico. Este relato tecnológico tem, portanto, como objetivo apresentar o desenvolvimento de solução do problema definido na área de tecnologia, referente ao atraso nas entregas de projetos no departamento de tecnologia do Senac – SP. A integração entre os dados coletados e os negócios agrega conhecimento à metodologia de gerenciamento de projetos ágeis e às mudanças estruturais nos times de tecnologia. Este relato tecnológico reforça a possibilidade de mudanças organizacionais mesmo em ambientes corporativos conservadores, reforçando o engajamento dos envolvidos.

PALAVRAS-CHAVE

DevOps. Metodologia ágil. Tecnologia da informação.

ABSTRACT

This technical report analyzes information technology (IT) project management methodologies and integration of teams in the corporate environment of the Serviço Nacional de Aprendizagem do Comércio de São Paulo (Senac – SP). The study was based on different project management methodologies, demonstrating their usefulness and applicability and the phases covered until the result's delivery. The adoption of a management model is based on a strong alignment with the strategic objectives of the business and the respective areas to obtain the best solution despite the gaps that can be identified during its adoption. A relative increase in delays in the delivery of technology projects and collaboration between

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

technology specialists was observed in the entity. Senac – SP has many projects in line for execution. The different assets and technological platforms demonstrate difficulties regarding operations, platform development, quality, and support, which can directly affect the delivery or support of a technological asset. Therefore, this report aims to present the development of a solution to the problem defined in the technology area, referring to the delay in project deliveries in the technology department of Senac – SP. Integrating collected data and business adds knowledge to agile project management methodology and structural changes in technology teams. This report reinforces the possibility of organizational changes even in conservative corporate environments, reinforcing the engagement of those involved.

KEYWORDS

DevOps. Agile methodology. Information technology.

INTRODUÇÃO

As metodologias de desenvolvimento de projetos de tecnologia da informação (TI) e as melhores práticas para gerenciamento e integração dos times de infraestrutura e desenvolvimento são consideradas por Veras (2014) fatores determinantes da maneira como os projetos tecnológicos são entregues, sejam eles de análise do prazo acordado, sejam eles de escopos definidos no início do projeto, que estão sempre em constante mudança.

Este artigo aplicado é um estudo sobre as metodologias de gestão de projetos de TI e integração das equipes no ambiente corporativo do Serviço Nacional de Aprendizagem do Comércio de São Paulo (Senac – SP), analisando a aderência das melhores práticas de gerenciamento. O estudo amparou-se em diferentes metodologias de gestão de projetos, gestão ágil e desenvolvimento e operações (*development and operations [DevOps]*), demonstrando as suas utilidades e aplicabilidade e as fases que são percorridas até o momento da entrega do resultado.

A adoção de um modelo gerencial tem por base um forte alinhamento aos objetivos estratégicos do negócio e às respectivas áreas em questão, tentando obter a melhor solução, não obstante as lacunas que possam ser identificadas durante a sua adoção.

Ao se analisar o cenário atual e a estrutura organizacional do Senac – SP foi possível observar um relativo aumento nos atrasos de entregas de projetos de tecnologia e na colaboração entre especialistas de tecnologia. Identificado o problema, para solucioná-lo, buscou-se entendê-lo, analisando as suas ocorrências, as estratégias para solução do problema e o plano de ação e seus desdobramentos e avaliando os resultados obtidos, juntamente com as propostas de solução. Este relato tecnológico tem, portanto, como objetivo apresentar o desenvolvimento de solução do problema definido na área de tecnologia, referente ao atraso nas entregas de projetos no departamento de tecnologia do Senac – SP.

A expectativa é a criação de valor para os clientes por meio de entregas pontuais e faseadas com a participação ativa de todos envolvidos, proporcionando, com isso, a diminuição de atrasos em projetos de TI, beneficiando tantos os clientes internos do Senac quanto os externos, devido à demanda de serviços de outros estados.

A metodologia utilizada foi a apresentada por Marcondes et al. (2017) para solução de problemas e/ou aproveitamento de oportunidades.

CONTEXTO INVESTIGADO

O Senac é uma instituição brasileira de educação profissional criada em 10 de janeiro de 1946 por um decreto-lei. É uma entidade privada com fins públicos que recebe contribuição compulsória das empresas do comércio e é administrada pela Confederação Nacional do Comércio.

O serviço e a empresa

O Senac está presente há mais de 70 anos em todos os estados do Brasil, em mais de 1850 municípios, e capacitou mais de 2 milhões de brasileiros por ano, de acordo com as informações apresentadas na página do Senac na internet (Senac, 2020). Trata-se de uma empresa de ensino de capital misto que está entre as 20 maiores instituições de ensino no Brasil e com, aproximadamente, 9 mil funcionários no estado de São Paulo.

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

Devido à abrangência nacional, o departamento de TI, conhecido como Gerência de Tecnologia da Informação (GTI), tem diversos projetos com diferentes soluções a serem desenvolvidos e implantados, tanto para os alunos, que são considerados os clientes finais, quanto para os departamentos internos do Senac – que têm soluções estratégicas e administrativas que precisam ser apoiadas e desenvolvidas –, que são os clientes internos. O departamento de tecnologia tinha mais de 300 projetos para o ano de 2020, assim como diversas outras soluções e melhorias que entram em uma fila para provisionamento de recursos e priorização de demanda para que sejam consideradas um projeto. Entendem-se como clientes externos os alunos e usuários de aplicações externas ao Senac e como clientes internos os funcionários que trabalham na empresa, seja na sede, em São Paulo, ou em unidades operacionais.

A gerência de tecnologia da GTI é composta por uma equipe de, aproximadamente, 120 funcionários contratados em regime da Consolidação das Leis do Trabalho (CLT) e colaboradores terceiros, divididos em diferentes equipes e especialidades.

Em 2020 havia um *backlog* de 303 projetos a serem contemplados, assim como 603 projetos de melhorias, que são considerados ajustes em projetos já entregues. Os projetos de melhorias são dinâmicos, adicionados a todo instante, sendo importante a velocidade da entrega para não aumentar ainda mais a fila de projetos.

DIAGNÓSTICO DA SITUAÇÃO-PROBLEMA

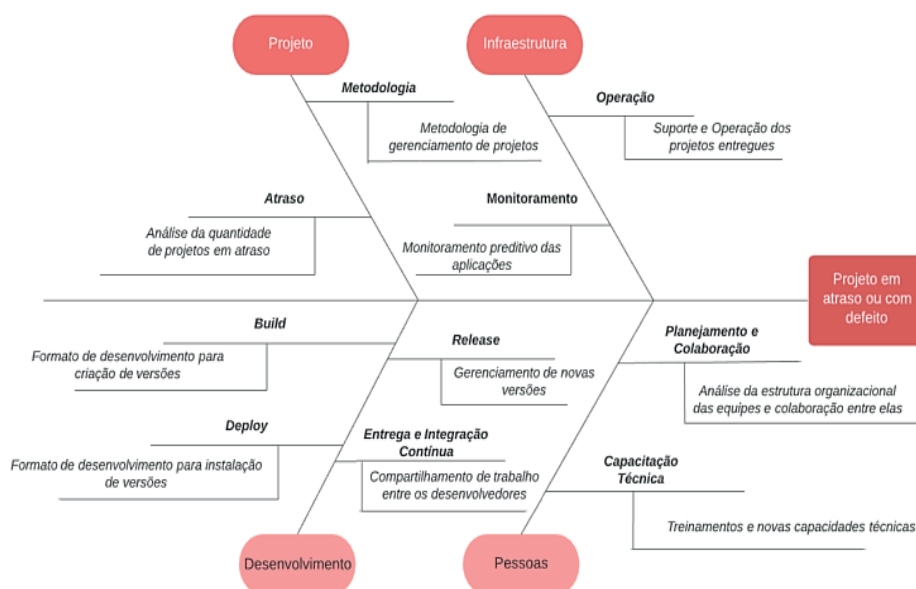
Para Marcondes et al. (2017), faz-se necessário aprofundar o entendimento da situação-problema. Um dos instrumentos utilizados para esse fim é o diagrama de Ishikawa, também conhecido como diagrama espinha de peixe, ou diagrama de causa e efeito.

De acordo com Ishikawa (1993), o agrupamento dos fatores deve ser corretamente controlado a fim de que os processos sejam transformados em causas que resultem em bons produtos e efeitos. Trata-se de uma ferramenta para analisar as causas de um determinado problema de maneira organizada, diminuindo as chances de algum problema ser esquecido. O diagrama auxilia na relação existente entre o resultado em não conformidade ou indesejado de um processo (efeito) e os diferentes fatores (causas) que podem contribuir para que

INTEGRAÇÃO COLABORATIVA E AGILIDADE EM PROJETOS DE TECNOLOGIA DA INFORMAÇÃO DO SENAC – SP

determinado problema ou resultado tenha ocorrido. Essa ferramenta caracteriza-se como um instrumento para controle de qualidade, aplicável em atividades diversas, pois contribui para a identificação de desvios no fluxo logístico, observando uma possível existência e localização dos gargalos na organização em que se aplica a ferramenta (Ishikawa, 1993).

FIGURA 1 – Diagrama de Ishikawa



Fonte: Elaborada pelos autores.

Com base em observações e análises de cada entrega e por dificuldades notórias em participações nos projetos, assim como a vivência no departamento de tecnologia, foi possível analisar e destacar as percepções e elaborar, conforme a Figura 1, os processos e as causas que resultam em um projeto com atraso ou defeito:

- **Projeto:** A metodologia utilizada, assim como a quantificação de projetos em atraso, a organização colaborativa para execução do projeto e o formato em que o planejamento é executado.

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

- **Metodologia:** Metodologia para gestão e condução de projetos.
- **Atraso:** Análise dos dados por ferramentas de *Business Intelligence* BI para visualização dos projetos em atraso e dentro do cronograma
- **Infraestrutura:** Entendem-se como infraestrutura os compostos tecnológicos, tais como servidores e armazenamento de dados.
- **Monitoramento:** Metodologia e ferramentas para monitoramento das aplicações e disponibilidade dos ambientes.
- **Operação:** Suporte das aplicações e funcionalidades, operação com foco na análise preditiva e na melhoria da disponibilidade.
- **Desenvolvimento:** O método utilizado pelos desenvolvedores e a maneira como são lançadas novas versões, assim como a entrega desses serviços e a integração entre eles.
- **Build:** Maneira como são geradas as versões dos projetos e a ferramenta utilizada para gerenciamento.
- **Deploy:** Formato em que são realizados os desenvolvimentos dos projetos para instalação das versões em ambientes de homologação, desenvolvimento e produção.
- **Release:** Ferramenta utilizada e metodologia para *releases* de versões das aplicações.
- **Entrega e integração contínua:** Metodologia e ferramentas para compartilhamento de trabalhos e rotinas entre os desenvolvedores.
- **Pessoas:** Os papéis e as responsabilidades dos colaboradores, analistas e desenvolvedores e a capacidade técnica requerida para as mudanças e novas tecnologias abordadas.
- **Capacitação técnica:** Análise das equipes quanto às melhorias propostas e aos conhecimentos tecnológicos e à eventual necessidade de treinamentos.
- **Planejamento e colaboração:** Análise da estrutura organizacional para verificação da necessidade de ajustes na estrutura organizacional, assim como acontece a colaboração entre as equipes dos projetos.

Para o diagnóstico do problema, também foram coletados dados dos projetos atribuídos ao ano de 2020 e foram realizadas análises das equipes de tecnologia e as suas respectivas divisões, onde, atualmente, é utilizado o *framework* do Itil (*Information Technology Infrastructure Library*), método destinado a organizar processos de TI e

orientar profissionais a exercerem suas funções com eficiência em projetos, governança de TI e governança corporativa.

De acordo com o PMBOK (PMI, 2017), “um projeto é um esforço temporário empreendido para criar um produto, serviço ou resultado exclusivo. Os projetos e operações diferem, principalmente, no fato que os projetos são temporários e exclusivos, enquanto as operações são contínuas e repetitivas”.

O departamento de tecnologia do Senac – SP tem em seu *backlog* (listagem de projetos aguardando execução) um total de 724 projetos, que são categorizados como:

- **Melhoria:** Projetos já implantados que precisam de ajustes, novas funcionalidades ou correções.
- **Novos projetos:** Projetos que precisam ser desenvolvidos e são considerados novas soluções ou entregas.

Pode-se, ainda, identificar também a quantidade de projetos e melhorias que precisam ser entregues, com o envolvimento de diversas equipes, sejam elas técnicas, divididas em equipes e especialização, ou de negócio, divididas em departamentos. As informações foram retiradas da ferramenta Project da Microsoft, que é a aplicação institucional para controle de projetos.

Para realização dos projetos, é seguido um modelo de distribuição departamental, dentro da GTI, utilizando o conceito de distribuição por especialidades e aplicações, ou seja, são separados por “silos”. Os departamentos, dentro das suas especialidades, são integrados e compartilham os mesmos projetos, ora como responsáveis, ora como participantes ou colaboradores, levando-se em conta, no momento do planejamento, o cronograma da equipe responsável pelo projeto.

Trata-se de um modelo tradicional, denominado cascata (*waterfall*), que deixa evidentes as etapas que os projetos precisam percorrer, pois precisam ser executadas de maneira sequencial, o que facilita a identificação da etapa na qual o projeto se encontra.

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

Fundamentação teórica para a proposta de solução

A solução oferecida é a adoção de uma metodologia amparada em um estudo de caso com os gestores das equipes de tecnologia que consiste em aperfeiçoar e melhorar o processo de gestão de projetos e as suas dependências tecnológicas, para possibilitar um trabalho conjunto, visando à melhoria de tempo de execução dos projetos, assim, evitando ou diminuindo o atraso nas suas respectivas entregas.

Para evitar o atraso nos projetos, o departamento de tecnologia deve formatar uma nova proposta para controle de cronograma de projetos, com participação mais efetiva dos departamentos envolvidos e novos formatos tecnológicos, permitindo a mudança na gestão e na condução dos projetos. Penrose (1959) considera que a singularidade de cada empresa está na distinção entre recursos e os possíveis serviços que podem ser obtidos com o emprego de cada um deles. Assim, a GTI do Senac atua na solução dos problemas, levando em conta as limitações institucionais e os arranjos departamentais, preparando o negócio para o novo modelo que agilize o desenvolvimento de projetos e que possibilite a gestão colaborativa entre os desenvolvedores e os operadores da tecnologia, propiciando uma entrega contínua das soluções.

A seguir, serão utilizadas técnicas consagradas para metodologias gestão ágil e DevOps, listando as que se aplicam à solução do problema proposto no trabalho.

Gestão ágil de projetos

O termo “ágil” está cada vez mais presente nas empresas e acaba sendo confundido com agilidade ou projetos mais rápidos. De acordo com Highsmith (2012), a agilidade é a habilidade de criar e responder às mudanças para conseguir obter lucro nos negócios. Reforçando o conceito, Kruchten (2008, p. 3), reconhecida referência em processos de desenvolvimento de *software*, define “ágil” como sendo “a habilidade que uma organização tem de reagir e se adaptar a mudanças no seu ambiente de forma mais rápida do que a taxa dessas mudanças”. De acordo com Camargo (2019, p. 100) “ágil” é “a habilidade

de entregar valor aos clientes, em um ambiente dinâmico, com incerteza, volatilidade, mudanças e adaptação constante”.

Adotar o modelo ágil, portanto, não significa fazer de forma rápida, mas ter um *mindset* ágil, isto é, deve-se ser efetivamente ágil. Quando se fala em *mindset*, quer-se referenciar um modelo mental pelo qual se compreende como as pessoas trabalham, como se relacionam e se comunicam entre si e como geram valor aos clientes. Essa dificuldade pode ser um fator determinante para a solução dos problemas, conforme reforçado novamente por Camargo (2019, p. 102), que afirma que “[...] não existe uma lista pronta de metodologias e ferramentas predefinidas para serem utilizadas em todas as situações”. Existem, porém, algumas práticas que podem ser adotadas para começar a gerenciar os projetos de forma ágil, como veremos mais adiante.

Essa metodologia vai de encontro ao atual modelo ora adotado baseado no controle e definido por etapas, ou seja, o modelo cascata, utilizado pela GTI, em que o foco da responsabilidade do departamento ou empresa é redirecionado de processos e ferramentas para a agilidade, que abrange não somente a rapidez, mas também chegar no lugar certo, com o escopo correto, no menor tempo possível.

Manifesto Ágil

No ano de 2001, um grupo formado por 17 especialistas em desenvolvimento de *software* se reuniu em Utah, nos Estados Unidos, para discutir uma nova forma de gerar melhores resultados em seus projetos. Eles buscavam uma alternativa ao modelo sequencial de desenvolvimento de *software*, como o ora adotado pela GTI, em formato de cascata e sequencial. Embora o Manifesto seja direcionado para a tecnologia, ele pode ser aplicado na empresa em diferentes departamentos e setores.

De acordo com a organização criadora da metodologia ágil (Agile Manifesto, s. d.), há quatro principais pilares:

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

1. **Indivíduos e interação entre eles, mais do que processos e ferramentas:** Fundamentado nos pilares da comunicação e interação entre os participantes do projeto, e não focando somente os processos e as ferramentas a serem utilizados.
2. **Software em funcionamento, mais do que documentação abrangente:** Foco na entrega e no funcionamento do *software*, dedicando maior atenção a entregas, e não somente à documentação.
3. **Colaboração do cliente, mais do que negociação de contratos:** Participação colaborativa desde o momento da criação do projeto, sendo a participação mais importante que as tratativas contratuais e os acordos entre áreas.
4. **Responder a mudanças, mais do que seguir um plano:** O escopo sofre mudanças a todo instante e um dos fatores determinantes será como se adaptará a essas mudanças e a velocidade em que será ajustado.

Princípios do Manifesto Ágil

Podem-se citar também os princípios do Manifesto Ágil, de acordo com a organização (Agile Manifesto, s. d.):

- Nossa maior prioridade é satisfazer o cliente pela entrega contínua e adiantada de *software* com valor agregado.
- Mudanças nos requisitos são bem-vindas, mesmo tardiamente no desenvolvimento. Processos ágeis tiram vantagem das mudanças visando a vantagem competitiva para o cliente.
- Entregar frequentemente *software* funcionando, de poucas semanas a poucos meses, com preferência por menor período.
- Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
- Construção de projetos em torno de indivíduos motivados, dando a eles o ambiente e o suporte necessários e confiando neles para a realização do trabalho.

- O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é a conversa face a face.
- *Software* funcionando é a medida primária de progresso.
- Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.
- Contínua atenção à excelência técnica e um bom *design* aumentam a agilidade.
- Simplicidade é essencial.
- As melhores arquiteturas, requisitos e *designs* emergem de equipes auto-organizáveis.
- Em intervalos regulares, a equipe reflete sobre como se tornar mais eficaz e, então, refina e ajusta seu comportamento de acordo.

Metodologia ágil: Scrum

O *Scrum*, sigla composta pelas letras iniciais das palavras em inglês “*Sprint*”, “*Cycle*”, “*Review*”, “*Update*” e “*Meeting*”, é um exemplo de metodologia ágil a ser adotada no desenvolvimento de *softwares* que se apoia nos princípios do Manifesto Ágil para elaboração e dinâmica do formato de trabalho e, de acordo com Vettorazzo (2018), é uma das metodologias mais utilizadas pelos times de desenvolvimento de software e a mais usada para exemplificar os conceitos do método ágil.

De acordo com Cohn (2011, p. 30), em um projeto *scrum*, testadores e programadores têm que desaprender alguns comportamentos. Os testadores precisam aprender que o teste deve abordar a conformidade e as necessidades do usuário, assim como os programadores precisam entender que não é necessário um projeto em sua totalidade para iniciar a codificação.

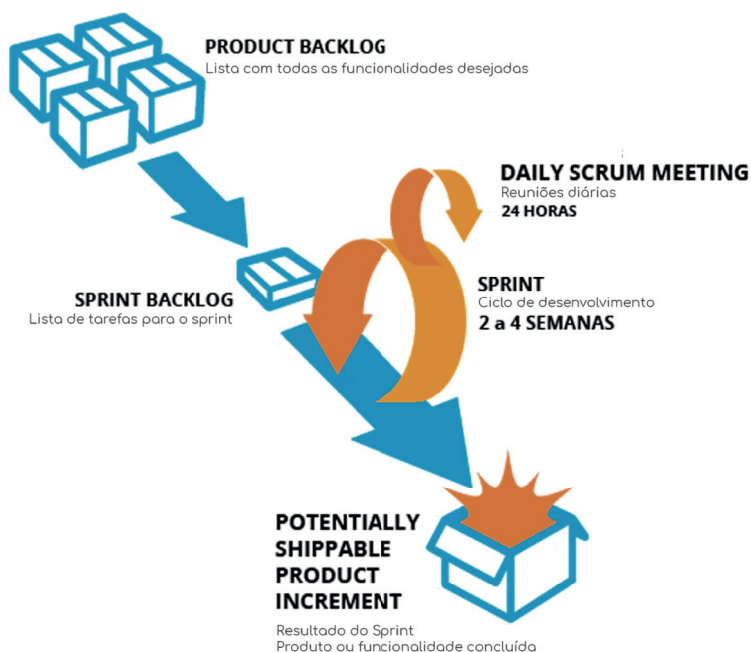
A metodologia abordada, o *scrum*, parte de um *product backlog*, que são as listas de coisas a se fazer, retirando dessa lista os itens que precisam ser desenvolvidos, e tudo isso acontecendo um ciclo predefinido de tempo. Os ciclos definidos são chamados de *sprints*, que são o resultado de um produto ou funcionalidade no período definido e

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

normalmente duram de duas a quatro semanas, também com reuniões diárias entre os times para verificação de problemas ou necessidade de ajuda e ajuste devido a qualquer problema que possa estar acontecendo no período.

No final dos *sprints*, após validados, o time entrega os itens, ficando liberado para receber novos produtos do *backlog*.

FIGURA 2 – Modelo ágil de desenvolvimento de *software* (*scrum*)



Fonte: Faberhaus, 2023.

Na Figura 2, é possível verificar o fluxo de entregas desde o *backlog* e o resultado das entregas, com as reuniões e *sprints*, até o momento da entrega de um produto ou funcionalidade.

De acordo com Vettorazzo (2018, p. 17), *scrum* é um *framework* no qual as pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com mais alto valor possível. É leve, simples de entender

e extremamente difícil de dominar. No *scrum*, os projetos estão divididos em ciclos e o *sprint* representa um *time box* no qual um conjunto de atividades devem ser executadas, sendo elas requisitos, análise, projeto, evolução e entrega.

Para Vettorazzo (2018, p. 19), quadro *scrum* é um instrumento que possibilita organizar o *backlog*, assim como acompanhar o andamento das tarefas que estão sendo trabalhadas no *sprint* atual. Desse modo, o processo pode ser acompanhado mediante *cards* que são movimentados e classificados como item de *backlog* “a fazer”, “sendo feito” e “feito”. Esses *cards* são movimentados conforme o cronograma ou a tarefa muda de etapa.

Existem diferentes ferramentas de *scrum* para auxiliar os times na criação dos *backlogs* e na criação de *cards* (figuras interativas de trabalho), que podem ser acessados pela internet e facilitam a troca de informações, ajudando as equipes a estruturarem e melhorarem seus fluxos de trabalho. Usualmente, utiliza-se um recurso visual para a organização dos *cards* e *sprints* denominado Trello. Nessa interface, os cartões são movimentados e podem ser designados para recursos específicos, com organização das datas até o momento da entrega, o que acontece de maneira faseada, caracterizado pelos *sprints*.

DevOps

Como mencionado anteriormente, o termo “DevOps” é a contração em inglês das equipes envolvidas na construção e na implantação de *software*. “Dev” é referente à equipe de desenvolvimento responsável pelos testes, pela codificação e pela análise do projeto (*development*), já “Op” se trata da equipe responsável pela solução de incidentes e problemas e pela criação e implantação do ambiente de produção e monitoramento (*operations*).

De acordo com Muniz et al. (2019), DevOps é uma cultura fortemente colaborativa entre as equipes de desenvolvimento e de operações para entregar *software* funcionando de forma ágil, segura e estável. Reforça-se também que, mais do que um conceito, é uma jornada de aproximação entre as pessoas com ações práticas de automação para acelerar as implantações com qualidade e empatia entre todos os envolvidos.

Quanto às equipes de TI, cita-se o exemplo desse trabalho do Senac: as equipes de tecnologia estão separadas em diferentes estruturas e departamentos e há conflitos sobre

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

de quem é a responsabilidade de um problema, quando desenvolvedores alegam problema de infraestrutura e a infraestrutura aponta problemas no desenvolvimento. Com o DevOps, munido das melhores práticas e ferramentas, os conflitos tendem a diminuir, visando à empatia pautada nos processos e soluções tecnológicos para melhoria contínua e processos enxutos.

Muniz et al. (2019) apontam as ações concretas para disseminar a cultura DevOps e promover os quatro pilares para a sua adoção efetiva:

1. **Colaboração:** Construção de resultado com interações de pessoas com experiências e um propósito em comum.
2. **Afinidade:** Relações interdependentes entre os times com objetivos organizacionais complementares para reforçar o sentimento de empatia entre as equipes.
3. **Ferramentas:** Aceleradores e impulsionadores de mudança cultural.
4. **Escala:** Dimensionamento que leva em conta como os outros três pilares podem ser aplicados.

INTERVENÇÃO PROPOSTA

Plano de ação

O plano de ação estabelece a estratégia adotada para diminuição dos projetos utilizando a gestão ágil de projetos e DevOps para integração das equipes e projetos, assim como as necessidades tecnológicas para implantação de acordo com os aspectos analisados.

Inicialmente, foi necessária a aprovação da GTI, feita mediante uma apresentação para exibir as necessidades e as lacunas existentes entre os times de infraestrutura e desenvolvimento, e foi solicitada a criação de uma equipe que representasse as mudanças estruturais em tecnologia, bem como a definição de papéis e responsabilidades que se podem cumprir utilizando colaboradores cedidos por diferentes coordenações para criação da equipe. Na demonstração para as equipes de tecnologia, foram demonstrados, também, os conceitos de DevOps e método ágil, assim como a quantidade de projetos que se encontravam em atraso.

Com a aprovação da diretoria, foi necessária uma apresentação para todo o departamento de tecnologia do Senac – SP com um plano contendo um *roadmap* com as principais entregas e planejamento com os macros dos principais pontos a serem cumpridos de maneira institucional.

Entre os principais riscos referentes às mudanças propostas, estavam a falta de conhecimento das metodologias e tecnologias apresentadas e a possível resistência a mudanças dos colaboradores no processo.

Para financiamento das atividades foram utilizadas somente soluções que já se encontravam disponíveis de maneira gratuita nas comunidades ou parcerias com empresas de *software* que já tinham vínculo com o Senac por relação contratual ou educacional.

Projeto “Metodologia Ágil e DevOps”

Na intervenção, inicialmente, foi criada por um dos autores deste artigo, como líder do projeto, uma equipe denominada Middleware, focada na infraestrutura, com o objetivo de ser uma facilitadora na comunicação com as equipes de desenvolvimento.

O motivo de não se criar uma equipe de infraestrutura com nomes utilizados atualmente, como DevOps, foi evitar resistência quanto a eventuais mudanças que poderiam acontecer, evitando-se ansiedade nas equipes de desenvolvimento, possibilitando, assim, que as necessidades e as facilidades que a equipe de infraestrutura propiciou fossem tratadas de maneira conjunta, com a participação de todos no processo, e que fossem pontos-chave para a implantação da cultura ágil e da metodologia DevOps.

Um dos cuidados durante todo o processo de mudanças estruturais e metodológicas foi com a cultura da empresa. Muitos colaboradores com muitos anos de empresa poderiam se sentir afetados pelo processo e por suas possíveis mudanças e consequências.

Todas as etapas a serem cumpridas e um *roadmap* com os objetivos a serem alcançados foram apresentados em uma reunião com a GTI. Após o término da exposição, muitos colaboradores procuraram um dos autores deste artigo, pois identificavam-se com o processo e estavam ansiosos por tais mudanças, pois precisavam também de ajuda nas tratativas com a equipe de infraestrutura e com algumas lacunas que existiam para

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

o desenvolvimento das soluções. Todas as etapas foram cumpridas e estão disponíveis, sejam em melhorias processuais ou tecnológicas.

O passo inicial para implantação da metodologia DevOps foi a retirada de acesso de todos os desenvolvedores aos servidores de produção dos aplicativos, ou seja, não mais poderiam alterá-los quando já em uso ou reiniciar serviços sem o consentimento de todos, assim como realização de testes e homologação nos ambientes produtivos.

Para executar essa tarefa, foram enviados formulários para preenchimento dos desenvolvedores para iniciar as etapas apresentadas no *roadmap* da equipe. Esse passo inicial foi considerado de extrema importância, pois erros repetitivos seriam descobertos e, com isso, iniciava-se a desoneração de tempo de desenvolvedores em funções que não são de sua responsabilidade, eliminando-se, assim, as múltiplas funções.

O DevOps do Senac foi configurado, aplicando as ferramentas de tecnologia à elaboração das etapas da metodologia. Um dos autores deste artigo elaborou e implantou na equipe liderada por ele todas as etapas consideradas tecnologicamente necessárias para o DevOps, sendo elas:

- **Codificação (*code*):** Foi utilizada a solução GitLab, que é um gerenciador de repositório de *software* baseado em Git, que é um gerenciador de versões de *software*.
- **Compilação (*build*):** Implantou-se a solução Maven, que é uma ferramenta de automação de compilação, e o Nexus, para gerenciar artefatos de *software*.
- **Teste:** Inicialmente foi utilizado o Apache Bench para testes manuais das aplicações e dos serviços.
- **Release:** Para provisionamento automático de versões (*releases*), foi disponibilizada a ferramenta Jenkins para administração. Por ela, são disparadas rotinas de atualização e versão das aplicações.
- **Deploy:** O Openshit e conceito de *container* para aplicações legadas e passíveis de migração.
- **Monitoramento:** A solução Dynatrace foi implantada pela equipe de Middleware e é utilizada para identificação de problemas e análise da experiência do usuário.

- **Operação:** Ansible é uma ferramenta de automação de rotinas de infraestrutura, para gerenciar, automatizar, configurar servidores e implantar aplicativos, por linha de código, solução incorporada no Openshift/Kubernetes.

RESULTADOS OBTIDOS

Todas as etapas percorridas auxiliaram na economia de licenciamento de *software*, em torno de R\$ 2.500,00 por licença, resultando em cerca de R\$ 700.000,00 de economia com a criação de mais de 300 serviços, que agora são executados em formato de *container* e não fazem uso de um sistema operacional licenciado, e na redução do número de incidentes com a retirada de acessos e utilização da ferramenta Dynatrace, pois todos os problemas foram tratados pontualmente.

Outro fator importante é quanto à disponibilidade dos serviços. No modelo de *container*, as mudanças são realizadas de forma que não tragam mais indisponibilidade e podem ser feitas de maneira contínua e integrada, agilizando o processo de gestão de mudança, não sendo mais necessária intervenção humana, podendo-se focar somente novas funcionalidades, assim como uma base de conhecimento criada para compartilhamento de informações e códigos para reaproveitamento de implantações e documentação. Todos os monitoramentos já são nativos no momento da criação dos *containers*, ou seja, não é necessária instalação, configuração de ferramentas de monitoramento manual.

Os serviços, por estarem sendo executados em formato de *container*, desoneraram a equipe de monitoramento 24x7 e os desenvolvedores com horas de trabalho, pois, em caso de falhas ou alto consumo, o ambiente ajusta-se automaticamente, realizando uma conta de homem/hora. É possível constatar que, somente com a exoneração de horas de um analista júnior (R\$ 9.000,00) e um programador pleno (R\$ 7.000,00), foi possível economizar algo em torno de R\$ 280.000,00 e mais horas da equipe de monitoramento, o que, até a elaboração deste trabalho, ainda não fora calculado, pois propicia-se a possibilidade de redução de colaboradores de missão crítica que atuam durante todos os períodos no monitoramento e em procedimentos para aplicações.

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

No referido trabalho, foram migradas mais de 300 aplicações para esse modelo apresentado, contemplando a estruturação do desenvolvimento e modelo de *container* e já existe a previsão da migração e criação de mais 300 serviços.

Com as metodologias adotadas para gestão de projetos, aliadas ao DevOps para integração das equipes e soluções tecnológicas, foi possível entregar projetos de maneira faseada para homologação dos demandantes, possibilitando um escopo flexível nos projetos e ajustes pontuais e momentâneos nas soluções, com isso, auxiliando nos prazos dos projetos e no cumprimento das datas, assim como na aceleração do desenvolvimento dos projetos em atraso.

CONTRIBUIÇÃO TECNOLÓGICA-SOCIAL

Neste artigo aplicado, foi possível aprofundar-se nas principais metodologias e soluções tecnológicas tanto com o uso de aplicativos para promover as mudanças como quanto à gestão de projetos e à integração entre as equipes. As mudanças continuam acontecendo na GTI e a proposta apresentada está sendo utilizada em outros projetos, mostrando resultados surpreendentes, com entregas dentro do prazo combinado e algumas até mesmo antes do período acordado.

A inovação proposta com a mudança no formato de desenvolvimento e na gestão de projetos é um passo muito grande da instituição e propicia inúmeras possibilidades tanto para o fornecimento de soluções aos clientes internos quanto aos externos, o que resulta em possibilidades de aumento de receita, mediante a comercialização de soluções que proporcionem aos clientes mecanismos para integrar equipes e soluções tecnológicas.

Durante essa jornada, muitos desafios foram enfrentados e um dos principais foi a rotatividade de funcionários e, apesar do esforço de retenção do Senac – SP, houve perda de alguns colaboradores da equipe, devido ao cenário da pandemia e à quantidade de soluções e ofertas pela internet e, também, devido ao fato de os colaboradores terem um perfil multidisciplinar de soluções e gestão de projetos, que é de grande relevância para o mercado de trabalho, sendo compreensível, tendo-se em vista a evolução profissional desses colaboradores, uma vez que o Senac – SP é uma entidade de capacitação.

Quando se faz a abordagem da metodologia ágil e do DevOps, muitos pensam que eles entram em conflito de alguma maneira, mas é possível observar que se complementam em pontos cruciais das entregas faseadas, em questões colaborativas entre as equipes, e o DevOps surge como um facilitador para que essas entregas aconteçam no âmbito tecnológico, apoiado em ferramentas estruturais e definidas para cada etapa de um processo de entrega de *software*.

Uma sugestão é que as mudanças possam ocorrer, mesmo quando se trata de cenários corporativos mais conservadores. Um dos pontos principais é reforçar a necessidade do engajamento de todos os envolvidos, assim como da equipe que organizará todas essas mudanças, pois, até o total entendimento e a capacitação das equipes, muitos conflitos e preocupações podem acontecer, mas, na verdade, até após o total entendimento do processo e dos benefícios propostos, pode haver desconfiança.

Um outro ponto importante que deve ser reforçado é com relação à *containerização*, que se mostrou uma grande aliada na automação de rotinas manuais, no aumento de disponibilidade das aplicações e no controle das mudanças. Quando se tem uma solução dessas para utilização, o caminho para integrar as soluções que facilitam o trabalho dos desenvolvedores é mais simples e desonera o tempo que muitos gastam com a administração dessas aplicações. Um outro ponto positivo é que devido ao fato de as aplicações estarem centralizadas em uma única solução, facilitam-se a gestão e a padronização do trabalho de todos os colaboradores.

Quando se trata de projetos ágeis com entregas faseadas de acordo com o modelo adotado, não é possível pensar nesse modelo sem pensar em uma mudança estrutural e tecnológica nos times de desenvolvimento de sistemas e, nesse trabalho, foi possível identificar que ambas se complementam e não conflitam nas necessidades, recomendações e solução dos problemas.

REFERÊNCIAS

Agile Manifesto. (s. d.). *Princípios por trás do Manifesto Ágil*. <https://agilemanifesto.org/iso/ptbr/principles.html>

MARCELO DINIZ AUGUSTO E ALBERTO DE MEDEIROS JR.

- Amazon Web Services (AWS). (2020). *O que é DevOps?* <https://aws.amazon.com/pt/devops/what-is-devops/>
- Camargo, R. (2019). *Project model visual: Gestão de projetos simples e eficaz*. Saraiva.
- Cohn, M. (2011). *Desenvolvimento de software com Scrum*. Sagah.
- Faberhaus (2023). *Metodologia ágil: O que é e onde o design entra nisso*. <https://www.faberhaus.com.br/metodologia-agil/>
- Fagundes, E. M. (2012). Um kit de ferramentas para excelência na Gestão de TI. *Data-Driven Business*. <https://efagundes.com/artigos/cobit/>
- Highsmith, J. (2012). What is agility? <https://www.thoughtworks.com/insights/blog/what-agility>
- Ishikawa, K. (1993). *Controle de qualidade total: À maneira japonesa*. Campus.
- Kruchten, P. (2008). *Agility situated: Contexto does matter, a lot*. The University of British Columbia.
- Marcondes, R. C., Miguel, L. A. P., Franklin, M. A., & Perez, G. (2017). *Metodologia para elaboração de trabalhos práticos e aplicados: Administração e contabilidade*. Editora Mackenzie.
- Muniz, A., Santos, R., Irigoyen, A., & Moutinho, R. (2019). *Jornada DevOps: Unindo cultura ágil, lean e tecnologia para entrega de software com qualidade*. Brasport.
- Penrose, E. G. (1959). *The theory of the growth of the firm*. Wiley.
- Senac. Site institucional do Serviço Nacional de Aprendizagem do Comércio. (2020). <https://www.senac.br>
- Veras, M. (2014). *Gerenciamento de projetos: Project model Canvas (PMC)*. Brasport Livros e Multimídia.
- Vettorazzo, A. (2018) *Engenharia de Software*. Porto Alegre: Sagah.