

SISTEMAS OPERACIONAIS DE REDES ABERTAS

Roque Maitino Neto



SOLUÇÕES
EDUCACIONAIS
INTEGRADAS



O papel do DevOps

Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Explicar conceitos de DevOps.
- Relatar a necessidade de DevOps.
- Exemplificar o uso de DevOps.

Introdução

Desenvolver um produto de *software* não é, definitivamente, algo simples. Essa tarefa exige, além de colaboração, a compreensão das necessidades de quem encomendou o *software* e o empreendimento de recursos adequados para a finalização satisfatória do projeto. Trata-se de desafios que as organizações enfrentam há décadas para o desenvolvimento de produtos que estejam de acordo com a demanda do cliente. Nesse sentido, o DevOps é uma prática que visa a tornar essa empreitada mais propensa ao sucesso do que ao fracasso, por meio da colaboração entre os times de desenvolvimento e de operações.

Neste capítulo, você vai estudar conceitos relacionados ao DevOps, verificando a desconstrução de alguns mitos e compreendendo quais são os objetivos dessa prática. Por fim, você vai explorar um guia para a implantação da prática.

1 O DevOps: conceitos e mitos

Assim que a tecnologia da informação (TI) se consolidou no ambiente corporativo como parte inseparável de um empreendimento, ela deixou de ser apenas uma facilitadora das atividades corriqueiras da empresa para se tornar uma força integradora entre departamentos, projetos e usuários. Como exemplos de suas aplicações, podemos apontar: um sistema de planejamento de recursos empresariais, que integra as várias funções e departamentos da empresa; uma *wiki* empresarial, que integra os múltiplos conhecimentos desenvolvidos

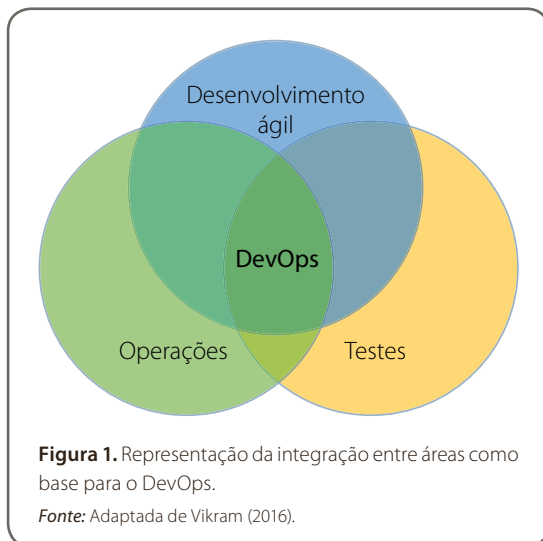
naquele ambiente; um servidor, que integra e permite o compartilhamento das atividades dos usuários. Embora os exemplos sejam inúmeros, a menção de alguns poucos já é suficiente para se perceber a importância da TI no ambiente empresarial.

Mesmo sendo o elo entre diversas atividades corporativas, até recentemente, faltava à TI um senso de integração entre as várias especializações que surgiram nas últimas décadas. Por ser um ramo de atividade bastante amplo — e ainda em expansão —, as áreas que têm a TI como “mãe” muitas vezes não apresentam vínculo entre si. Por exemplo, um especialista em redes pode não ter habilidade em entender como funcionam as aplicações para *web*, e alguém que ocupe seu tempo desenvolvendo programas pode não saber o que se passa com as operações relacionadas ao gerenciamento da infraestrutura de uma empresa. Graças ao DevOps, esses aspectos vêm mudando nos últimos anos.

DevOps é uma combinação de dois termos em inglês: *development* (desenvolvimento) e *operations* (operações). Ele é definido como uma cultura ou prática que enfatiza a colaboração e a comunicação dos desenvolvedores de *software* e de outras equipes de TI, automatizando o processo de entrega do *software* e as alterações na infraestrutura. A abordagem do DevOps visa a estabelecer uma cultura e um ambiente que incentivem a construção, o teste e o lançamento de produtos de qualidade de maneira mais rápida e confiável (THORPE, 2020).

Em um artigo publicado em seu *site*, a Amazon Web Services (O QUE É..., c2020) se refere ao DevOps como a combinação de filosofias culturais e de práticas que aumentam a capacidade de uma empresa de distribuir aplicativos e serviços em alta velocidade. Ainda segundo a AWS, essa combinação permite ao setor de TI otimizar e aperfeiçoar produtos em um ritmo mais acelerado do que se usasse processos tradicionais de desenvolvimento de *software* e gerenciamento de infraestrutura (O QUE É..., c2020).

A integração entre desenvolvimento de *software* e gerenciamento de infraestrutura parece ser, então, o coração do DevOps. Em sua definição, a UpGuard (c2020) destaca essa união entre áreas: o DevOps se refere a um movimento emergente entre profissionais de TI que defende uma relação de trabalho colaborativa entre desenvolvimento e operações de TI, resultando em um fluxo rápido de trabalho, que aumenta a confiabilidade, a estabilidade e a resiliência do ambiente de produção. A Figura 1 ilustra esse conceito.



Dentre os motivos que justificaram a criação dessa prática, sabe-se que a falta de integração entre times de desenvolvimento e de operações certamente foi o principal deles. Thorpe (2020) afirma que, no passado, as equipes de desenvolvedores e operações trabalhavam em silos com diferentes objetivos e prioridades. Se considerarmos o ciclo de vida de produção de um *software* e os papéis tradicionais nele presentes, veremos que a equipe de desenvolvimento será a responsável por conceber e produzir o *software*, mas quase nunca por tratar do armazenamento e da infraestrutura de comunicação do mesmo, por exemplo. Por outro lado, a equipe de operações deve garantir que o produto entregue ao cliente esteja estável, funcione o mais rapidamente possível e que seja tão amigável quanto o cliente solicitou.

Embora os objetivos dos desenvolvedores e operadores pareçam convergir, os meios que ambos utilizam para os atingirem podem ser divergentes. Por exemplo, os operadores estão normalmente interessados na entrega de um sistema estável, enquanto os desenvolvedores desejam que o produto chegue ao mercado o mais rapidamente possível. Em um cenário mais conturbado, as operações e o desenvolvimento são times adversários; as atividades do pessoal de teste e as atividades do pessoal de segurança são empreendidas apenas no final de um projeto, tarde demais para corrigir quaisquer problemas encontrados. Esses fatos não apenas podem contribuir para tempos de execução extremamente longos, mas também para a queda na qualidade do trabalho, resultando em impactos negativos para os negócios (KIM *et al.*, 2017).

Esse contexto de divergência entre equipes certamente ajudou a promover a ideia de integração, tão bem adotada e aprimorada pelo DevOps. Há, no entanto, algumas ideias preconcebidas — que podemos chamar de mitos —, que podem colocar gestores em dúvida e conduzi-los a uma situação de hesitação quando precisarem decidir sobre a adoção ou não da prática. A primeira delas, segundo Kim *et al.* (2017), é a de que colocar o DevOps em prática significa eliminar por completo as operações de TI. Trata-se de uma interpretação incorreta da ideia do DevOps. Embora a natureza do trabalho de operações de TI possa mudar, ela permanece tão importante como sempre. No DevOps, desde as fases iniciais do ciclo de vida, o pessoal de operação colabora com o pessoal de desenvolvimento, que, por sua vez, continua a trabalhar com o pessoal de operações muito tempo depois da liberação do produto. Com essa associação, fica claro que, em um ambiente DevOps, a colaboração se inicia nas fases embrionárias do projeto e continua mesmo após a implantação do *software*.

Outra associação direta que deve ser desfeita é a de que o DevOps substitui a metodologia ágil de desenvolvimento (*Agile*) ou que ambas não poderiam conviver em um ambiente de desenvolvimento. Na verdade, os princípios e as práticas do DevOps são compatíveis com o *Agile*, que geralmente serve como um facilitador eficaz da prática, devido ao seu foco em equipes pequenas, que liberam código de alta qualidade aos clientes, regularmente. A documentação da Microsoft corrobora essa ideia:

A coisa mais importante a saber sobre o DevOps *versus* o Agile é que eles não são mutuamente exclusivos. O DevOps é uma cultura que promove a colaboração entre todos os participantes envolvidos no desenvolvimento e na manutenção do *software*. O Agile pode ser descrito como uma metodologia de desenvolvimento criada para manter a produtividade e promover versões com a realidade comum de necessidades em constante evolução. Embora o DevOps e o Agile sejam diferentes, quando usados juntos, essas duas metodologias levam a uma maior eficiência e resultados mais confiáveis (DEVOPS..., c2020, documento *on-line*).

Mesmo que a integração seja a sua meta principal, já que vincula pessoas, processos e tecnologias entre si, há outras conquistas a serem atingidas quando se opta pelo DevOps. A seguir, serão apresentados os objetivos e princípios da prática.

2 Objetivos e princípios do DevOps

Como toda prática que envolve melhorias em processos, o DevOps também possui seus objetivos. Os objetivos gerais de qualquer iniciativa de implantação do DevOps são sintetizados no Quadro 1.

Quadro 1. Objetivos do DevOps

Pessoas	Processos	Tecnologias
Nesta dimensão, espera-se mudança cultural efetivada em colaboração, cooperação e transformação de atitudes, além de melhor retenção de conhecimento e compreensão mais profunda do negócio. Espera-se que as pessoas adquiram maior capacidade de resposta às necessidades do empreendimento.	Em relação aos processos, espera-se, com o DevOps, que novos processos sejam mais bem aceitos. Além disso, busca-se aprimoramento da qualidade nas implantações de <i>software</i> , por meio da redução do número de incidentes e problemas por versão. Neste escopo, busca-se também lançamentos de versões de <i>software</i> mais frequentes, em consequência do aumento da velocidade de liberação de produtos para produção. Por fim, busca-se visibilidade aprimorada dos processos e requisitos de TI.	Em relação à tecnologia, o DevOps busca melhor qualidade do código, o que impacta na redução de defeitos em cada nova versão da aplicação. Outro objetivo nesta dimensão de tecnologia é o desenvolvimento mais ágil de produtos.

Fonte: Adaptado de UpGuard (c2020).

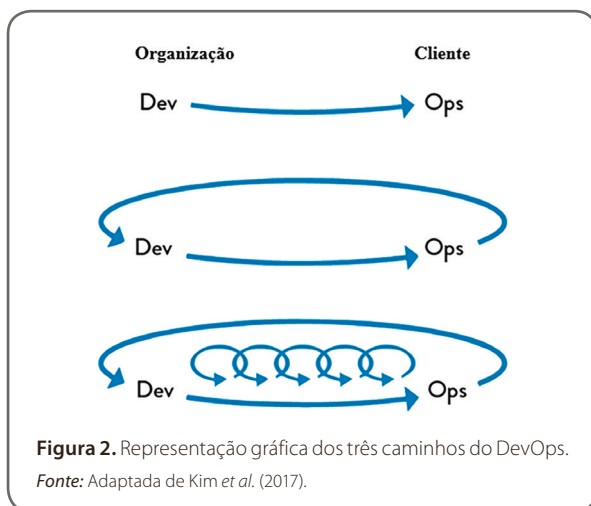
Embora o quadro nos revele apontamentos importantes sobre o que se pretende com a implantação do DevOps, uma forma mais eficiente de entender seus objetivos é por meio da compreensão de como seria um processo de desenvolvimento de *software* ideal com a aplicação do DevOps. Suponha um cenário em que os patrocinadores do *software* e o pessoal de desenvolvimento, qualidade, operações e segurança trabalhem juntos, visando ao apoio mútuo e ao sucesso geral da organização. Ao trabalharem vinculados por um objetivo único, eles direcionam seus esforços (cada um à sua maneira) para a codificação de centenas de linhas de código por dia, alcançando estabilidade, confiabilidade, disponibilidade e segurança em níveis bem altos. Nesse cenário, as equipes multifuncionais experimentam com afinco todos os possíveis recursos que tornarão mais rica a experiência dos usuários, sem perderem de vista as metas que aquele produto deve atingir em prol da corporação. Embora as equipes se preocupem com a implementação de recursos voltados ao usuário, elas não descuidam da fluidez de seu trabalho, garantindo, assim, o mínimo de interrupções nas operações de TI ou de qualquer outro cliente interno ou externo. De forma simultânea e colaborativa, o controle de qualidade, as operações e o pessoal da segurança empreendem esforços para reduzirem eventuais atritos nas equipes, criando sistemas de trabalho que permitem aos desenvolvedores serem mais produtivos. A junção dos saberes e das habilidades das equipes deixa o ambiente propício para que as organizações implantem um sistema de trabalho em que pequenas equipes possam desenvolver, testar e implantar códigos de maneira rápida e independente (KIM *et al.*, 2017).

Nesse cenário imaginado, é bastante natural supormos que os objetivos do DevOps relacionados a pessoas, processos e tecnologia serão atingidos de uma maneira fluida e confortável. Colocando na mesma perspectiva as situações relatadas nesse “mundo ideal” e os objetivos descritos no Quadro 1, podemos compreender como as práticas ideais de DevOps beneficiam o ambiente de trabalho.

- **Pessoas:** a mudança cultural (quase sempre necessária para que se possa trabalhar em grupo e de forma integrada) e a melhor compreensão do negócio são consequências do trabalho multidisciplinar proposto pelo DevOps e relatado nas situações com as quais acabamos de ter contato. Abastecidas com informações vindas de todas as equipes envolvidas e com um sentido aguçado de comunicação, as pessoas terão maior capacidade de resposta às necessidades dos negócios e às mudanças de rumos que certamente acontecerão no processo de desenvolvimento.

- **Processos:** um processo de desenvolvimento adequado não é aquele que trata as mudanças (entendidas em sentido amplo) como algo excepcional e indesejável. Ao contrário, um processo de *software* deve acomodar as mudanças e processá-las com agilidade. Além disso, um processo construído sobre a base forte da integração entre equipes tende a provocar redução no número de incidentes e problemas em produtos já lançados. Por fim, a velocidade de liberação de produtos em ambiente de produção tende a ser maior.
- **Tecnologias:** em um cenário de afinidade entre equipes, comunicação em nível elevado e troca de saberes, o reflexo será sentido na melhor qualidade do código e em um desenvolvimento mais ágil. Com uma boa diversidade de habilidades frequentando a mesma equipe, o uso da tecnologia apropriada no projeto será uma certeza.

Nesse cenário, é difícil imaginar que a produtividade do desenvolvedor não atinja seu ponto máximo, que não haja compreensão aprofundada do negócio e que cliente e equipe não terminem satisfeitos com seu projeto. Para que o cenário de mundo ideal do DevOps não fique apenas como um objetivo distante e dificilmente atingível, foram desenvolvidos princípios que sustentam essa prática. Kim *et al.* (2017) apresentam os princípios do DevOps, identificando-os como um conjunto de **três caminhos** (do original *three ways*) sobre os quais se apoiam todos os comportamentos e padrões relacionados ao DevOps. A Figura 2 mostra a representação gráfica do *three ways*.



O primeiro caminho torna possível um fluxo rápido de trabalho da esquerda para a direita, do desenvolvimento para as operações e para o cliente. A maximização do fluxo implica em tornar o trabalho mais visível, por meio da redução dos lotes de trabalho e da redução dos intervalos de trabalho. Ainda de acordo com Kim *et al.* (2017), o segundo caminho permite o fluxo rápido e constante de *feedback* da direita para a esquerda em todas as etapas do fluxo. Ele exige que o *feedback* seja aprimorado, para impedir que os problemas aconteçam novamente ou, em caso de ocorrência, que tenham detecção e recuperação mais rápidas. Esse procedimento cria qualidade na fonte e gera conhecimento onde é necessário, o que permite criar sistemas de trabalho cada vez mais seguros, em que os problemas são encontrados e corrigidos muito antes de ocorrer uma consequência séria.

O terceiro caminho permite a criação de uma cultura de alta confiança, que dá suporte a uma cultura de experimentação, em que a ocorrência de um eventual erro, em vez de ser motivo de “condenação” do membro do projeto, promove aprendizado. Diminuindo-se e ampliando-se continuamente os ciclos de *feedback*, são criados sistemas de trabalho cada vez mais seguros, o que habilita a equipe a assumir riscos e realizar experimentos. O terceiro caminho também cria um ambiente de trabalho em que a equipe pode multiplicar os efeitos de novos conhecimentos, transformando descobertas locais em melhorias globais. Independentemente de onde alguém realiza um trabalho, ele o faz com a experiência acumulada e coletiva de todos na organização (KIM *et al.*, 2017).

3 Um guia para a aplicação do DevOps

Embora os conceitos e os objetivos do DevOps, estudados anteriormente, sejam úteis para o início das discussões, não há garantia de sucesso de uma iniciativa de DevOps sem que algumas providências sejam tomadas e que alguns conhecimentos sejam adquiridos pelas equipes. A UpGuard (c2020) propõe uma série de ações que visam a viabilizar, de fato, a aplicação do DevOps em uma organização. Essa sequência tem início com o levantamento dos objetivos do negócio, conforme será descrito a seguir.

Entender quais são os objetivos da organização

Os objetivos do negócio em que o DevOps deverá ser inserido devem ser levantados e compreendidos. Cabe registrar aqui que os objetivos do DevOps e os objetivos do negócio não são a mesma coisa. Antes de mergulhar a fundo na sua iniciativa DevOps, um gestor deve ter o maior número possível de conversas com os colegas de negócios (incluindo a liderança executiva), a fim de que possa entender melhor quais são as iniciativas estratégicas específicas de negócios, quais são os principais desafios e pontos de atrito atualmente, além, é claro, de conhecer as necessidades e os desejos gerais do time de TI. Esse levantamento preliminar de informações ajudará a tornar a iniciativa de DevOps mais eficaz a longo prazo.

Via de regra, as iniciativas de levantamento de informações sobre um empreendimento envolvem, entre outros meios, reuniões, entrevistas, aplicação de questionários e pesquisas no *site* da empresa, sempre com o objetivo de formar entendimento sobre o que é relevante para as finalidades do empreendimento. Nesse sentido, a seguir, são exemplificadas algumas perguntas que podem ser feitas nessas conversas.

- “Para garantir que não tenho noções incorretas ou preconcebidas, você poderia me dizer exatamente o que faz aqui nesta empresa?”
- “O que diferencia um bom dia de um dia ruim para você?” (Aqui, a intenção é extrair os principais pontos problemáticos do cenário do “dia ruim”).
- “Quais são as suas metas para este ano e quais delas correm maior risco de não serem atingidas?”

É importante que as perguntas ajudem a entender melhor as principais necessidades da empresa e as suas estratégias.

Ter consciência da situação da empresa e prestar atenção em suas variações

Imagine a seguinte situação: a versão final de uma aplicação funcionou muito bem em todo o procedimento de testes, mas, uma vez implantada, não correspondeu às expectativas. É natural que uma pergunta seja feita: “O que fizemos na fase de testes que não funcionou agora?”. O levantamento de quais sistemas e ativos de TI a organização possui (incluindo dispositivos físicos,

sistemas virtualizados e instâncias de nuvem híbrida) é fundamental para a implementação bem-sucedida do DevOps.

A falta de consciência situacional adequada foi identificada como uma das principais causas de situações como a que descrevemos. Depois de levantar os objetivos e desejos da organização, o pessoal de implantação do DevOps deve adquirir visibilidade completa de departamentos e nichos, para garantir a entrega mais confiável e repetível possível dos serviços de TI. Ter visibilidade abrangente, precisa e atualizada é a base para uma tomada de decisão bem-sucedida e sem surpresas. Assim, a iniciativa de implantação do DevOps requer a descoberta dos dispositivos de TI da organização, das suas configurações e das mudanças que eles podem sofrer a qualquer momento.

Uma das principais vantagens do DevOps é o alinhamento e a colaboração que ele pode criar. Ao estabelecer uma imagem clara do estado de toda a infraestrutura da organização, será possível estabelecer um diálogo produtivo com as várias entidades envolvidas e empreender ações em prol da adoção da prática. O Quadro 2 resume as ações tomadas em relação a essas entidades envolvidas no projeto.

Quadro 2. Ações a serem tomadas para garantir a adoção do DevOps em uma organização, de acordo com a entidade ou o departamento

Entidade (ou departamento)	Ação
Desenvolvimento de aplicações	Integração das ferramentas de implantação, para garantir o provisionamento preciso e automatizado de compilações no laboratório de teste e no ambiente de produção.
Operações de TI	Garantia de que as ferramentas de suporte de serviço e rastreamento de erros nas operações e no desenvolvimento permaneçam sincronizadas, para melhor rastreamento de problemas, melhor suporte ao cliente e melhoria contínua dos serviços.
Garantia da qualidade	Teste do desempenho do aplicativo em ambientes de laboratório e produção, visando a acelerar as correções e reduzir o tempo médio de reparo.
Segurança e auditoria	Aproveitamento de um conjunto comum de ferramentas, para reduzir o risco à segurança e melhorar os relatórios de conformidade.

(Continua)

(Continuação)

Quadro 2. Ações a serem tomadas para garantir a adoção do DevOps em uma organização, de acordo com a entidade ou o departamento

Entidade (ou departamento)	Ação
Fornecedores	Esclarecimento das responsabilidades dos fornecedores, por meio de um acordo de nível de serviço.
Clientes	Aprimoramento da experiência do cliente com implantações e ciclos de <i>feedback</i> mais rápidos, enquanto se reduz o desperdício de tempo em cada ciclo.

Fonte: Adaptado de UpGuard (c2020).

Depois que a visibilidade é criada entre as principais partes interessadas, é essencial entender as lacunas e avaliar todas as mudanças que estão acontecendo nesses sistemas. O passo seguinte está relacionado à definição de processos, conforme descrito a seguir.

Definir claramente processos e partes interessadas

A definição das responsabilidades ao longo do ciclo de vida de um produto é fator decisivo para o sucesso de qualquer iniciativa de DevOps. A multiplicidade de pontos de contato e integrações significa que a tarefa de alinhar, priorizar e executar os requisitos de todas as partes interessadas é um desafio. Antes da efetiva implantação das práticas de DevOps, é essencial primeiro definir claramente os processos e as partes interessadas relevantes, assim como garantir o apoio dos principais interessados desde o início. A comunicação com os envolvidos em termos que lhes são familiares contribuirá para aumentar a aceitação e a visibilidade e, muitas vezes, acelerar uma iniciativa de DevOps.

Estabelecer meios de medição

A efetivação deste item significa encontrar uma forma de medir os resultados da iniciativa de DevOps, para tomar conhecimento do sucesso ou não do projeto. O sucesso geralmente está parametrizado em algum marco, e não em uma análise real do impacto e dos resultados obtidos. A medição, portanto,

é sempre muito importante para uma iniciativa de DevOps, já que tal iniciativa tem como alvo os próprios processos que determinam como o trabalho de TI é feito. Não poder medir é um problema, mas não saber o que medir pode ser muito pior.

No âmbito da TI, há uma grande variedade de ferramentas que realizam algum tipo de medição, o que pode aumentar a chance de aquisição de alguma que não sirva ao propósito estabelecido. Embora sejam necessários monitoramento de *log*, infraestrutura, configuração, aplicativo e rede, nenhum deles seria tão útil sem a clara definição do que se deseja medir. O sucesso de uma iniciativa de monitoramento de TI não pode ser medido em termos de dados coletados ou de métricas aplicadas. Ele só pode ser medido em termos de sua capacidade de influenciar as decisões de negócios de maneira positiva.



Exemplo

Este exemplo aborda o caso de uma organização (nome não revelado) que passava por uma atualização importante em sua plataforma. Para ter sucesso nesse projeto, ela precisava adotar os principais princípios do DevOps, para dimensionar e implantar serviços rapidamente. Mediante a orientação da empresa Stackoverflow, foram implantadas as práticas do DevOps que viabilizam entrega contínua por meio da criação, do teste e da liberação das aplicações com muito mais frequência e garantia. Como resultado da transformação dos processos da organização, todos os envolvidos na atualização se tornaram proprietários do projeto e responsáveis pelo objetivo final. Os desenvolvedores passaram a ter um compromisso com códigos de maior qualidade, e o pessoal de operações passou a ser responsável por cuidar de incidentes e garantir a boa experiência do cliente. Não há mais silos de informações e trabalho, e as equipes compartilham prontamente sucessos e falhas. A simplificação do processo de DevOps faz com que as equipes se tornem mais autônomas e responsáveis pelo sucesso da operação (DEVOPS..., 2016).

A adoção do DevOps, portanto, não é empreendida em uma só ação e sem planejamento prévio. Em vez disso, requer levantamento de informações, conhecimento das necessidades do empreendimento, definição dos processos que conduzirão à prática e, por fim, estabelecimento de formas de medição.



Referências

DEVOPS case study: streamlining DevOps principals through automation. *In*: STACKOVERDRIVE. New York, 16 apr. 2016. Disponível em: <https://www.stackoverdrive.net/devops-case-study-streamlining-devops-principals-automation>. Acesso em: 3 jun. 2020.

DEVOPS versus Agile. *In*: MICROSOFT Azure. Seattle, c2020. Disponível em: <https://azure.microsoft.com/pt-br/overview/devops-vs-agile/>. Acesso em: 3 jun. 2020.

KIM, G. *et al.* *The DevOps handbook: how to create world-class agility, reliability, and security in technology organizations*. Portland: IT Revolution Press, 2017.

O QUE É DevOps? *In*: AWS. [S. l.], c2020. Disponível em: <https://aws.amazon.com/pt/devops/what-is-devops/>. Acesso em: 3 jun. 2020.

THORPE, E. *DevOps: advanced methods and strategies of using DevOps*. [S. l.: s. n.], 2020.

UPGUARD. *Four prerequisites for DevOps success*. São Paulo: Apple Books, c2020. Disponível em: <https://books.apple.com/br/book/four-prerequisites-for-devops-success/id1081166048>. Acesso em: 3 jun. 2020.

VIKRAM, K. *Introduction to DevOps*. [S. l.: s. n.], 2016.



Fique atento

Os links para sites da web fornecidos neste capítulo foram todos testados, e seu funcionamento foi comprovado no momento da publicação do material. No entanto, a rede é extremamente dinâmica; suas páginas estão constantemente mudando de local e conteúdo. Assim, os editores declaram não ter qualquer responsabilidade sobre qualidade, precisão ou integridade das informações referidas em tais links.

Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.

Conteúdo:



SOLUÇÕES
EDUCACIONAIS
INTEGRADAS