



DevOps na prática

Apresentação do DevOps como um conjunto de práticas de engenharia de software, processos ágeis e governança de tecnologia da informação, gerando valor para o cliente final com o menor tempo e maior qualidade dos produtos digitais.

Prof. Ubirajara Mendes Nunes Junior

Propósito

Desenvolver soluções de software e manter o poder de inovação é fundamental para qualquer pessoa ou empresa sobreviver atualmente. O grande desafio é como garantir fluidez de entrega de valor para os clientes. É justamente para garantir essa fluidez de entrega de valor e ritmo de inovação que implantar uma abordagem DevOps é essencial para as empresas.

Preparação

Instale o Visual Studio Comunidade, disponível para download.

Objetivos

- Reconhecer a plataforma Azure e seus principais serviços ofertados.
- Identificar o Azure DevOps e como suas ferramentas elevam o nível de qualidade dos softwares desenvolvidos.
- Implementar um exercício simulado de entrega de funcionalidade de um sistema web na plataforma Azure DevOps.

Introdução

Imagine que tenhamos a oportunidade de iniciar uma jornada juntos e que ao final dela precisemos entregar uma solução para um problema conhecido, no qual o desenvolvimento de um produto digital é fundamental e tenha que ser feito o mais rápido possível e sem erros. Mas, para isso, eu preciso conhecê-lo melhor para confiar em você. Então, vou começar te fazendo alguns questionamentos.

Se eu te perguntar quão produtivo você é, qual seria a sua resposta? E se eu for além e te perguntar como você consegue medir essa produtividade? E se eu te questionar em relação à qualidade do resultado da sua produtividade, ou seja, a sua entrega. O que você me apresentaria? Essas são perguntas que não possuem respostas fáceis, porém elas são a base para a maioria dos times que trabalham juntos no desenvolvimento de produtos de software.

Nesse contexto, a utilização de uma abordagem DevOps pode ajudar você a conseguir tais respostas. Neste conteúdo, você entenderá por que esse conjunto de práticas e técnicas estão revolucionando o modo como as empresas se organizam a fim de entregar inovação e valor para seus clientes, de forma rápida, com alta qualidade e frequente.

Atualmente, a prática de uma abordagem DevOps é uma realidade na maioria das empresas, e aquelas que ainda não estão aplicando, inevitavelmente o farão em algum momento no futuro. Por isso, conhecer bem esses conceitos será fundamental para o seu desenvolvimento profissional. Estar preparado para engajar em uma empresa que já adota essa prática, ou até mesmo auxiliar empresas em sua adoção, certamente dará a você um diferencial no mercado.

Agora que você entende bem a importância de DevOps, vamos iniciar essa jornada!



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Cloud computing

Confira agora os principais aspectos sobre os modelos de cloud computing.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

O que é?

Antes de falarmos sobre Microsoft Azure, é importante pontuarmos alguns conceitos sobre o que é a computação em nuvem, quais seus diferenciais e benefícios em contraste ao modelo tradicional de infraestrutura de tecnologia.

Cloud computing, ou computação em nuvem, é a oferta de serviços de tecnologia da informação — tais como máquinas virtuais (VMs), serviços de armazenamento, bancos de dados e serviços de aplicação — por meio da internet, e o modelo de pagamento desses serviços se dá sob demanda, ou seja, você paga apenas pelo que consumir dos recursos contratados.

Hoje em dia os serviços ofertados pelas principais plataformas evoluem de maneira muito rápida e recursos avançados já são ofertados como serviços para internet of things (IoT), machine learning (ML) e artificial intelligence (AI).



Saiba mais

Internet of things (IoT): É uma rede de objetos físicos capaz de reunir e de transmitir dados. Machine learning (ML): É um ramo da inteligência artificial (IA) e da ciência da computação que se concentra no uso de dados e algoritmos para imitar a maneira como os humanos aprendem, melhorando gradualmente sua precisão. Artificial intelligence (AI): É uma área dentro da ciência da computação responsável por simular a inteligência e o comportamento humano usando apenas máquinas.

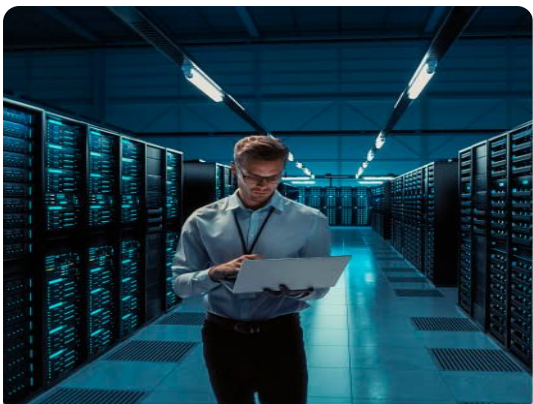
Agora vamos entender quais as diferenças entre o modelo tradicional de infraestrutura de serviços de TI e o modelo de cloud computing!

Modelo tradicional

A oferta de serviços de computação no modelo tradicional consiste na criação e manutenção de seus datacenters (centro de dados) nas empresas. Nesse modelo, a área de tecnologia deve se preocupar em analisar os recursos de computação necessários para manter suas operações de tecnologia que suportam o negócio, tanto do ponto de vista de software como de hardware.

No modelo tradicional, a empresa é responsável por construir, adquirir, gerir, manter, atualizar e expandir sempre que necessário toda essa infraestrutura, o que normalmente é caro e muito demorado.





Não é difícil entender que estamos apresentando um cenário complexo e desafiador, porém não se engane, essa ainda é a realidade para uma boa parte das empresas do mercado.

Algumas dessas empresas mantêm esse tipo de infraestrutura realmente por alguma necessidade específica do seu negócio, tal como confidencialidade dos dados, porém a maioria das empresas tem a oportunidade de reavaliar sua estratégia de tecnologia e quem sabe se beneficiar do modelo de cloud computing.

Modelo de cloud computing

Assim como no modelo tradicional, a área de tecnologia das empresas deve se preocupar em analisar os recursos de computação necessários para manter suas operações de tecnologia que suportam o negócio, porém muito mais voltado para as soluções de software do que para a infraestrutura de hardware onde essas soluções serão executadas. Isso porque, diferentemente do modelo tradicional, a cloud computing oferta tanto soluções de software quanto soluções de infraestrutura de hardware.

No modelo de cloud computing, as empresas não precisam se preocupar em construir, adquirir, gerir, manter, atualizar e expandir seus datacenters. Todo esse trabalho já está abstraído pelos provedores de serviços cloud e são ofertados como um conjunto de serviços que podem ser facilmente contratados. Assim, permite que as empresas foquem muito mais o seu negócio do que a manutenção da infraestrutura de tecnologia.

Isso significa que as empresas não precisam mais se preocupar com os serviços de TI que suportam o seu negócio?

Claramente não, porém todo o trabalho de se manter uma infraestrutura própria é drasticamente reduzido, além de garantir uma grande otimização desses custos e principalmente a agilidade em se expandir quando necessário.



Serviços de cloud computing

Assista agora para saber mais sobre os serviços de cloud computing, incluindo benefícios e modelos.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Benefícios da cloud computing

O tempo tem provado que a adoção de cloud computing é sem dúvida um caminho sem volta. A facilidade de se contratar serviços computacionais de maneira ágil, customizável e a um custo extremamente competitivo tem levado milhares de empresas a migrarem dos modelos tradicionais para a cloud computing, e já é extremamente comum que empresas que estão sendo criadas já nasçam com sua estratégia de tecnologia totalmente voltada para a cloud computing.

Poderíamos listar dezenas de benefícios que a cloud computing oferece que vem garantindo sua adoção, porém vamos focar os principais. Vamos lá!

Elasticidade

O conceito de elasticidade está relacionado à capacidade de se adicionar ou remover recursos computacionais sob demanda.

Imagine que você seja um profissional que trabalhe com edição de vídeos e precisa comprar um computador novo. Obviamente você fará uma análise sobre os principais softwares que utiliza e, baseado nisso, escolherá um computador com os recursos de processador, memória e armazenamento adequado para os seus objetivos.

Com o passar do tempo você adquire novas habilidades e demanda uso de novos softwares de edição de vídeo que, por sua vez, são mais especialistas e consequentemente demandam mais recursos computacionais. Você percebe, então, que seu computador fica muito lento e algumas edições requerem horas e horas de processamento e, por isso, deseja fazer com que esse tempo seja reduzido.



Comentário

O que você normalmente faria nessa situação? Simples! Você precisaria atualizar o seu computador com os recursos de processador, memória e armazenamento adequados para a sua nova realidade, o que vai requerer um novo investimento (que não é barato). Mas, além de ser um investimento alto, todo novo poder computacional fica boa parte do tempo ocioso, pois não é sempre que você está fazendo uma edição tão especializada. Como, então, equilibrar essa balança?

Não seria perfeito se você conseguisse um computador que adicionasse poder computacional quando você precisasse para a edição de vídeo especializada e diminuísse esse acréscimo de poder computacional quando você terminasse, pagando apenas pelo excedente do uso?

Esse é o benefício da elasticidade da cloud computing, você pode adicionar e remover recursos computacionais a qualquer tempo e paga apenas pelo uso desses recursos.

Agilidade

Vivemos em um mundo extremamente volátil, o que nos obriga a ter agilidade para abraçar as mudanças constantes. Além disso, precisamos a todo momento garantir que estamos fazendo entregas de valor para nossos clientes, o que requer alto poder de inovação.

Como vimos, quando uma empresa trabalha com um modelo tradicional de infraestrutura de tecnologia, nem sempre alterações ou adequações nessa infraestrutura conseguem ser tão rápidas, o que pode ser um problema para o negócio.

Ao adotar um modelo de cloud computing, conseguimos ter agilidade para mudar e inovar. Constantemente os provedores estão ofertando novos serviços, que se pode adotar de maneira rápida, e até mesmo a entrega de novos produtos das empresas se dá de maneira muito mais ágil.

Otimização de custos

O benefício de se pagar apenas pelo poder computacional consumido é um dos grandes fatores que vem impulsionando a mudança do modelo tradicional de infraestrutura para a cloud computing.

Façamos um breve exercício! Quando uma empresa possui um datacenter próprio, ela tem o custo de manter o local refrigerado de maneira adequada, o custo de energia para manter os servidores funcionando, além dos custos de aquisição, manutenção, e o principal, o custo do tempo ocioso dos equipamentos. Não seria ótimo se pudéssemos desligar o datacenter ou até mesmo desligar alguns servidores quando não estivessem em uso? Seria, e é possível, porém não é uma tarefa simples e expõe a empresa a muitos riscos, sendo o principal desses eventos o fato de um dos sistemas não ser reestabelecido.



Saiba mais

É nesse ponto que a cloud computing se destaca, nós só pagamos pelo recurso computacional que utilizamos. Caso uma solução, ou até mesmo parte dela, não precise ser utilizada, ela pode ser simplesmente desligada e religada quando necessário, de forma simples e prática. Por exemplo, para empresas que desenvolvem software é muito comum termos diferentes ambientes para desenvolvimento, homologação e produção. Durante um dia útil normal de trabalho, é esperado que todos os ambientes estejam prontos e disponíveis, mas o que dizer dos finais de semana? Não seria interessante desligar os ambientes de desenvolvimento e homologação para otimização de custos? Sim, e é totalmente possível e recomendado!

Modelos de serviços de cloud computing

Agora que você sabe o que é cloud computing e conhece alguns de seus benefícios, vamos conhecer os modelos de serviços que são oferecidos pela cloud computing.

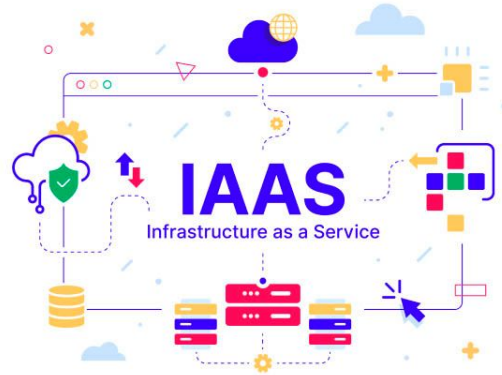
Cada um possui uma característica e um modelo de compartilhamento de responsabilidades com os clientes, veja:



Modelo de compartilhamento de responsabilidades

Infrastructure as a Service – IaaS

Infraestrutura como um serviço, ou IaaS, é o modelo de serviço mais básico da cloud computing e consiste na disponibilização de recursos computacionais básicos como redes virtuais, máquinas virtuais, serviços de armazenamento, entre outros.



Nesse modelo, o cliente é responsável por fazer toda a configuração da infraestrutura requerida bem como manter o ambiente seguro e atualizado, ficando sob a responsabilidade do provedor cloud a manutenção do hardware, conectividade (com a internet) e infraestrutura e segurança física.

O modelo IaaS de serviço foi o precursor da cloud computing e ainda é muito utilizado, porém, com tantas evoluções que os serviços cloud sofreram ao longo de tempo, é recomendada a utilização de outros modelos para arquiteturas mais modernas de aplicações.

Platform as a Services – PaaS

Plataforma como um serviço, ou PaaS, é o modelo de serviço que oferta uma camada superior dos recursos computacionais, removendo do cliente a necessidade de administração e manutenção do ambiente, tarefa que passa a ser do provedor.



Nesse modelo, as empresas focam o uso do recurso computacional não se preocupando com a infraestrutura. Por exemplo, se a empresa precisa publicar uma aplicação web na internet, ela não precisa se preocupar em criar uma máquina virtual, instalar o sistema operacional, manter, configurar para expor a internet para só então publicar sua aplicação. Basta que a empresa faça a contratação de um serviço de hosting de aplicações web do provedor cloud compatível com a stack de tecnologia com a qual sua aplicação foi desenvolvida, publicar a aplicação no serviço e pronto, a aplicação estará no ar!

Software as a Service – SaaS

Software como um serviço, ou SaaS, é o modelo de oferta de serviço no qual o cliente não se preocupa com nenhuma outra coisa a não ser utilizar a ferramenta disponibilizada pelo provedor de serviço cloud.



Quer um exemplo bem simples? Você simplesmente cria a sua conta no seu provedor de e-mail, seja ele Gmail ou Outlook, e começa a usar, sem nenhuma outra preocupação.

Plataforma Microsoft Azure

Neste vídeo, abordaremos os principais assuntos sobre o Microsoft Azure, incluindo o Azure Portal e os serviços da Azure.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

O que é Microsoft Azure?

Microsoft Azure é a plataforma global de serviços em nuvem oferecida pela Microsoft.

Ela contempla mais de 200 produtos e serviços para suporte das operações de tecnologia das mais diversas indústrias.

Microsoft Azure oferta os mais avançados serviços de computação em nuvem, contemplando todos os principais modelos de serviços IaaS, PaaS e SaaS. Além disso, oferta serviços avançados que servem de base para IA, ML e IoT, e a maioria dos serviços ofertados são cobrados no modelo Pay-as-you-go, ou seja, você paga pelo que foi utilizado do serviço contratado.

É uma das principais provedoras de serviço cloud do mercado, sendo considerada uma das líderes do segmento pela **Gartner**.



Gartner

O Gartner Group é uma empresa de TI, porém seu objetivo é servir de norteadora para as outras empresas desse segmento, em todo o mundo. Em suas atividades, destacam-se pesquisas, eventos, consultorias e prospecções que envolvem o mercado de TI.

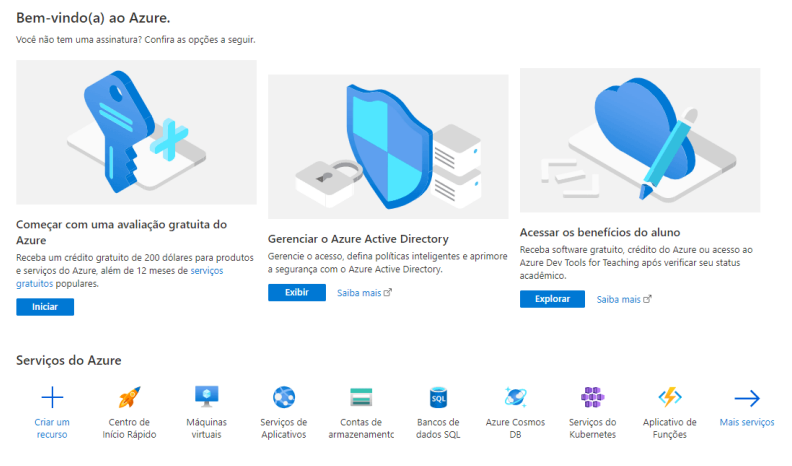
Vamos destacar algumas das principais categorias de serviços ofertados pela Microsoft Azure! Para mais informações, recomendamos o acesso ao site da Azure Microsoft.

Portal Azure

É o principal canal de interação com a Azure.

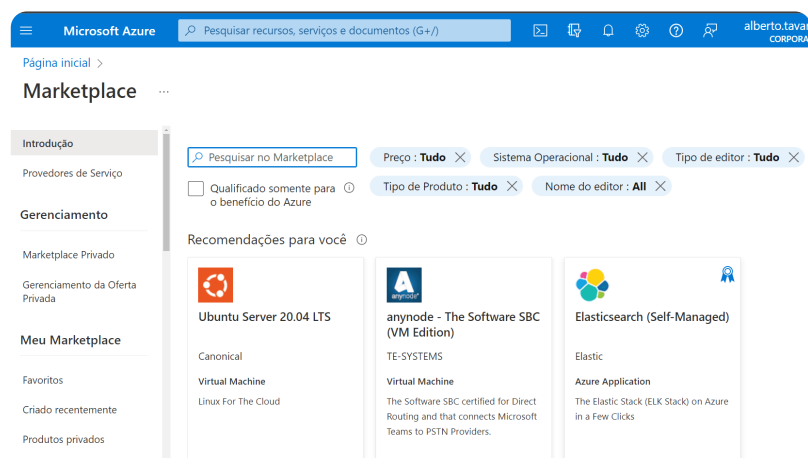
Por meio dele você pode gerenciar sua subscrição, provisionar os recursos computacionais necessários para a sua solução, gerenciar seus custos, configurar, monitorar e escalar esses recursos, tudo por meio de um portal web, simples, objetivo e customizável.

O portal Azure oferece ainda recursos que o ajudam a monitorar e otimizar suas aplicações, além de ter uma inteligência de analisar o comportamento dos recursos provisionados e sugerir melhorias na configuração para otimizar recursos e custos. Ele é acessado por meio do site portal.azure.com.



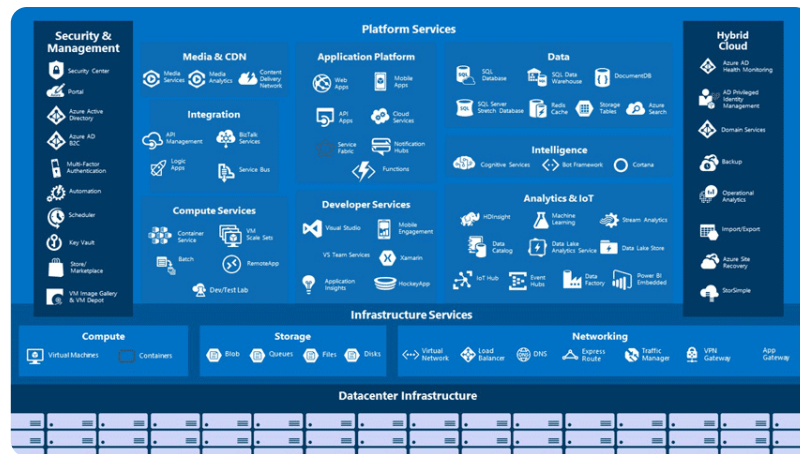
Azure Marketplace

É o canal de oferta de soluções de empresas que tem seus produtos e serviços otimizados para serem executados na Microsoft Azure. Toda solução ofertada no Azure Marketplace é certificada pela Microsoft.



Serviços da Azure

A Azure oferta mais de 200 tipos de serviços de computação em nuvem, vale destacar os serviços mais utilizados. Veja na imagem!



Compute Services

Serviços de computação são serviços de suporte a execução de aplicações na Azure desde seu build (compilação), hosting (hospedagem) até o deployment (publicação). Entre os principais serviços de computação podemos destacar:

Virtual Machines (VMs)

Permite a criação de máquinas virtuais, ou seja, um conjunto de recursos computacionais capazes de emular um hardware físico onde se pode instalar qualquer sistema operacional e, por sua vez, aplicações nesse sistema operacional.

Azure Container Services

Permite a criação de uma solução de hospedagem (hosting) de containers otimizada para ser executada na Azure. É utilizada para orquestrar e escalar aplicações conteneurizadas utilizando tecnologias como Docker Swarm ou Kube Azure Compute Services.

Azure Functions

Permite a criação de serviços de execução de códigos por demanda, independentemente do provisionamento de servidores e infraestrutura. É o serviço baseado no conceito de execução serverless (sem servidores).

Web e Mobile Services

Os serviços web e mobile permitem a criação de serviços de back-end para aplicativos web e mobile capazes de serem consumidos por qualquer aplicativo móvel independente se executados nas plataformas iOS, Android e Windows, de maneira rápida e fácil. Nesta categoria, destacam-se os seguintes serviços:

Azure App Service

Permite a compilação, hospedagem e execução de aplicações web desenvolvidas em qualquer linguagem, sem a necessidade de gerenciamento da infraestrutura necessária para sua execução.

Azure Mobile Apps

Permite a criação e execução de aplicações móveis altamente escaláveis e disponíveis globalmente. Entre as principais características que facilitam o desenvolvimento das aplicações móveis estão a sincronização de dados off-line armazenados no dispositivo, conectividade com dados locais, envio de notificações push e escalonamento automático dos serviços por demanda.

Databases

Permite a criação e execução de bancos de dados, relacionais e não relacionais, capazes de armazenar qualquer volume de dados e escalar de maneira rápida para atender altas demandas transacionais das aplicações. Podemos destacar os principais serviços:

Azure SQL Database

Banco de dados relacional totalmente gerenciado com escalabilidade automática e alto padrão de segurança. O Azure SQL Database oferece todo o poder do Banco de Dados Microsoft SQL Server como uma plataforma, sem a necessidade de gerenciamento da infraestrutura necessária para sua execução.

Azure Cosmos DB

Banco de dados não relacional com poder de distribuição global. O Azure Cosmos DB oferta diferentes modelos de bases de dados não relacionais, como DocumentDB, MongoDB, Apache Cassandra, PostgreSQL etc.

Verificando o aprendizado

Questão 1

A computação em nuvem permite acessar arquivos e executar diferentes tarefas pela internet, ou seja, você não precisa instalar aplicativos no seu computador para tudo, pois pode acessar diferentes serviços on-line para fazer o que precisa, já que os dados não se encontram em um computador específico, mas sim em uma rede. Nesse contexto, analise as afirmativas a seguir.

I. A empresa é responsável por construir, adquirir, gerir, manter, atualizar e expandir, sempre que necessário, toda infraestrutura de TI.

II. A área de tecnologia das empresas deve se preocupar em analisar os recursos de computação necessários para manter suas operações de tecnologia que suportam o negócio, porém muito mais voltado para as soluções de software do que para a infraestrutura de hardware onde tais soluções serão executadas.

III. A empresa não precisa se preocupar em construir, adquirir, gerir, manter, atualizar e expandir seus datacenters.

IV. A área de tecnologia deve se preocupar em analisar os recursos de computação necessários para manter suas operações de tecnologia que suportam o negócio, tanto do ponto de vista de software como de hardware.

Assinale a alternativa cujas afirmativas se relacionam com computação em nuvem:

A

Apenas as opções I, II e III.

B

Apenas as opções II, III e IV.

C

Apenas as opções III e IV.

D

Apenas as opções I, IV e V.

E

Apenas as opções II e III.



A alternativa E está correta.

O armazenamento de dados é feito em serviços que poderão ser acessados de qualquer lugar do mundo, a qualquer hora, não havendo necessidade de instalar programas ou armazenar dados, nem investir em hardware. O acesso a programas, serviços e arquivos é remoto, por meio da internet.

Questão 2

Em qual modelo a empresa contrata uma capacidade de hardware que corresponde à memória, armazenamento e processamento, por exemplo, podendo entrar nesse pacote de contratações os servidores, roteadores, racks, entre outros?

A

IaaS

B

PaaS

C

IoT

D

SaaS

E

Azure



A alternativa A está correta.

A forma de nuvem de hospedagem IaaS pode ser utilizada por clientes empresariais para criar soluções de TI facilmente escaláveis, nas quais as complexidades e despesas de gestão do hardware subjacente são terceirizadas para o provedor de nuvem.

Vamos começar!

Agora que conhecemos um pouco melhor o que é a plataforma de serviços em nuvem Microsoft Azure, vamos conhecer um serviço que tem como objetivo reunir pessoas, processos e tecnologia por meio da automação de entrega de software, a fim de prover uma entrega de valor contínua para os usuários, o Azure DevOps.

O Azure DevOps nos permite criar pipelines, que são conjuntos de tarefas bem definidas com o objetivo de garantir uma entrega de software para nossos usuários, de maneira automatizada e contínua. Entre essas tarefas, podemos integrar repositórios, automatizar o provisionamento de infraestrutura de serviços, compilar, testar e publicar artefatos de software, além de integrá-los de maneira simples e automatizada. Tudo isso relacionando as funcionalidades entregues naquele artefato com itens de backlog de produto que as identificam com o objetivo de garantir o rastreamento e auditoria de como, quando e por quem uma funcionalidade foi desenvolvida e entregue, esse é o poder do Azure DevOps.

O Azure DevOps é um conjunto de serviços que possibilita a automação de tarefas com o objetivo de automatizar a entrega de software, gerando valor continuado para o usuário.

Vamos conhecer agora quais são esses serviços e seus objetivos!

Azure Boards

Características do Azure Boards

Confira este vídeo sobre as características do Azure Boards! Nele, destacaremos conceitos de Agile e Scrum.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

O Azure Boards é a solução do Azure DevOps para gerenciamento das tarefas a serem desenvolvidas de seus produtos de software. É por meio do Azure Boards que será possível identificar as funcionalidades a serem desenvolvidas, que se desdobram em tarefas a serem medidas e distribuídas ao longo do tempo. Essas tarefas garantem histórico do desenvolvimento, pois armazenam as informações do que foi desenvolvido, por quem, em que momento e quanto tempo levou para ser desenvolvido.

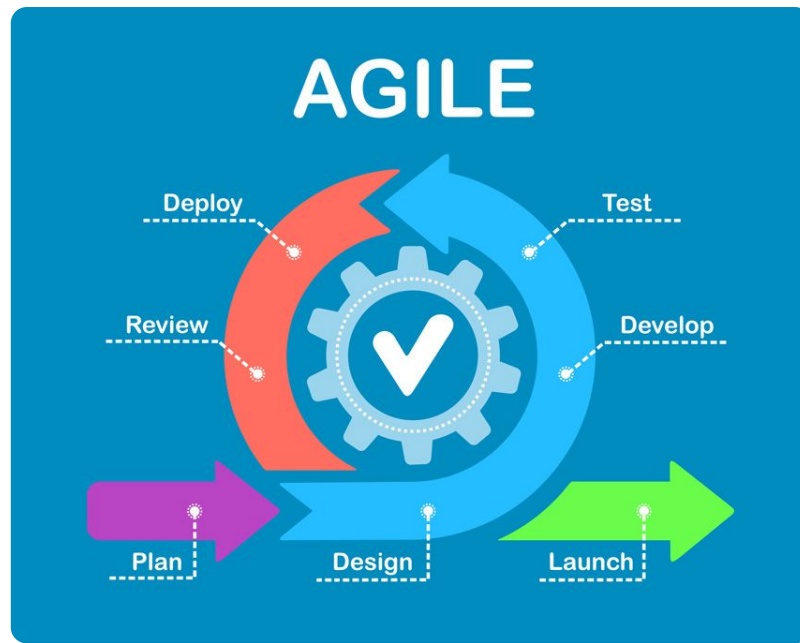
O Azure Boards provê um conjunto de templates nativos pré-definidos para suporte aos processos de gerenciamento do desenvolvimento baseado em Agile, Scrum e **Kanban**. O Azure Boards é flexível e customizável, característica que o torna adaptável a qualquer modelo de gerenciamento definido pelos times, além de prover um conjunto de ferramentas que ajuda no gerenciamento do desenvolvimento, como dashboards e ferramentas de relatórios.

Kanban

Disponibiliza dados atuais de cada projeto, permitindo um maior controle dos processos e das atividades, potencializando a produtividade, pois encurta caminhos e torna cada execução mais direcionada.

Agile

Azure Boards oferece um template de gerenciamento de projeto baseado nos conceitos de desenvolvimento ágil. Para se aprofundar nos conceitos de agilidade, você pode visitar o site da Agilealliance.

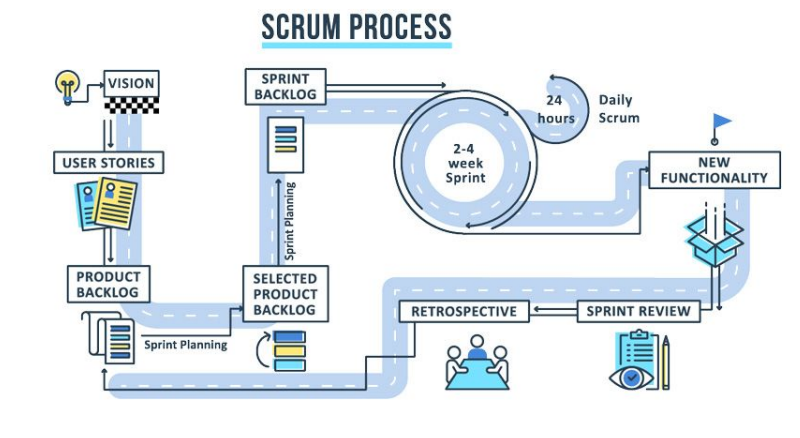


Ao criar um board baseado no template ágil, é possível criar itens de trabalho do tipo épico, funcionalidade, história de usuário, tarefas, erros e problemas.

Scrum

Assim como o template ágil, Azure Boards também oferece um template de gerenciamento de projeto baseado em Scrum. Para saber mais sobre os conceitos do Scrum, visite o site [scrum.org](https://www.scrum.org).

Ao criar um board baseado no template Scrum, é possível criar itens de trabalho do tipo épico, funcionalidade, product backlog item (pbi), tarefas, erros e impedimentos.



Azure Pipelines

Características e funcionalidades do Azure Pipelines

Neste vídeo, apresentaremos o Azure Pipelines, incluindo integração contínua, entrega contínua, Azure Test Plans e Azure Artifacts. Assista!



Conteúdo interativo

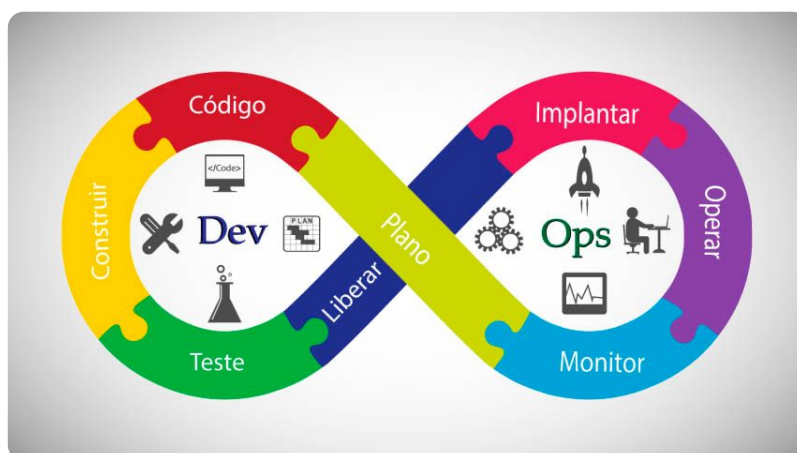
Acesse a versão digital para assistir ao vídeo.

Apresentação

Azure Pipelines é um orquestrador que permite a automação de tarefas e suporta a entrega contínua de software desde sua fase de compilação, passando por teste até sua publicação e execução. O Azure Pipelines suporta praticamente qualquer tipo de linguagem de programação e tipo de projeto de software existentes, atualmente, e combina ambos os conceitos de integração contínua (Continuous integration – CI) e entrega contínua (Continuous delivery – CD).

Integração contínua (Continuous integration – CI)

É a prática de engenharia de software usada pelas equipes de desenvolvimento para automatizar e agilizar a integração dos diferentes artefatos de softwares desenvolvidos pelos membros do time. Além dessa integração, tão logo o artefato desenvolvido é integrado à base existente, todo o código deve ser testado de maneira automatizada para rapidamente se identificar erros e inconsistências logo no início do ciclo de desenvolvimento, com o objetivo de manter o mais alto nível de qualidade possível. Os artefatos são produzidos a partir de sistemas de CI e são a base para serem publicados por meio do processo de entrega contínua.



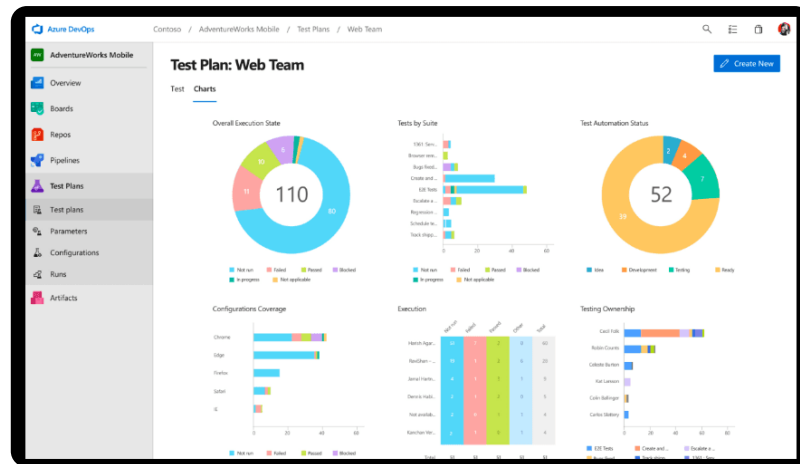
Entrega contínua (Continuous delivery – CD)

É a prática de engenharia de software usada pelas equipes de desenvolvimento para automatizar e agilizar o processo de publicação de um artefato de software gerado pela integração contínua em um ou mais ambientes de teste e produção para garantir a qualidade do que está sendo entregue ao usuário. O processo de entrega contínua é responsável por governar as diferentes versões dos artefatos de software que estão publicadas nos diferentes ambientes, seja de desenvolvimento, homologação ou produção.



Azure Test Plans

É uma suíte poderosa e avançada de ferramentas que habilita colaboração do time a fim de impulsionar a qualidade do software desenvolvido. O Azure Test Plans é uma solução de gerenciamento de testes que provê capacidades de testes manuais e automatizados dos diversos tipos, como unitários, integração, aceitação e exploração. Também possui uma plataforma de colaboração, onde membros de outros times da organização podem colaborar com o time de desenvolvimento, com feedbacks das funcionalidades utilizadas por eles.



Azure Artifacts

É uma plataforma de compartilhamento de componentes de software que permite aos desenvolvedores compartilharem seus componentes desenvolvidos por meio de **feeds**. O Azure Artifacts permite compartilhar os componentes com o time, com a organização e até mesmo publicamente. Também suporta o consumo de outros sistemas de gestão de artefatos como NuGet, NPM, Python, Maven.

feeds

Os feeds de artefatos são construções organizacionais que permitem armazenar, gerenciar e agrupar seus pacotes e controlar com quem compartilhá-los.

Repositórios Azure

Características dos repositórios Azure

Confira agora os principais aspectos sobre as características dos repositórios Azure, Git e TFVC.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

É um conjunto de ferramentas de controle de versão de código que deve ser utilizada para gerenciar a base de código do software desenvolvido. Assim como qualquer sistema de controle de código, o Azure Repos disponibiliza capacidade de rastreamento das alterações de código, reversão de alterações na base de código, entre outras. Azure Repos disponibiliza dois tipos de sistemas de controle de versão, Git e Team Foundation Version Control (TFVC).



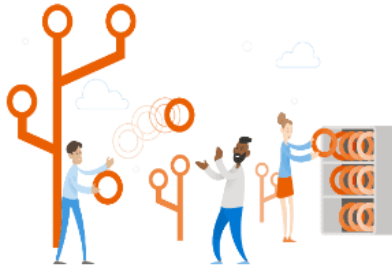
Comentário

O controle de versão evita que você tenha várias cópias de código em seu computador. Isso é perigoso, pois é fácil alterar ou excluir um arquivo na cópia errada do código, potencialmente perdendo trabalho. Os sistemas de controle de versão resolvem esse problema gerenciando todas as versões do seu código, mas apresentando uma única versão por vez.

A seguir, veja a tela de login do Azure Repos:



Azure Repos
Sign in

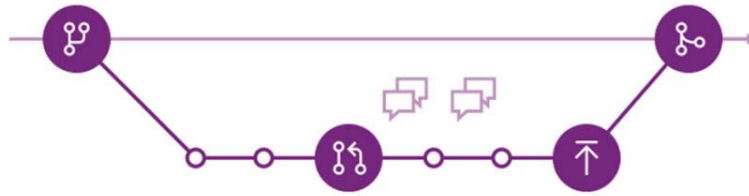


Git

É um sistema distribuído de controle de versão de código, ou seja, toda cópia da base de controle de código carrega consigo todo o histórico de alterações que o software sofreu ao longo do tempo. A vantagem desse modelo é que é possível fazer todas as alterações locais e facilmente sincronizar com uma base principal em um servidor. Git é o repositório padrão do Azure Repos.

Vejamos as etapas do fluxo de trabalho de controle de versão:

- 1) Obtenha uma cópia local do código se ainda não tiverem uma;
- 2) Faça alterações no código para corrigir bugs ou adicionar novos recursos;
- 3) Depois que o código estiver pronto, disponibilize-o para revisão por sua equipe;
- 4) Depois que o código for revisado, mescle-o na base de código compartilhada da equipe.



Team Foundation Version Control (TFVC)

É um sistema centralizado de controle de versão de código, ou seja, diferente do Git, uma cópia da base de controle de código carrega apenas a última versão das alterações que o software sofreu desde seu último sincronismo. Nesse modelo, sempre que um arquivo começa a ser alterado por um membro do time, esse arquivo é bloqueado no servidor e somente ao final da alteração e sincronismo volta a ser disponível para edição. Os dados históricos são mantidos somente no servidor.

Vejam as duas funcionalidades que merecem destaque:

Check-out

Permite que o usuário reserve, travando, o arquivo. Com isso, só ele pode realizar alterações naquele momento. Outros usuários podem ter acesso ao arquivo para o modo leitura, mas não para o modo escrita.

Check-in

Permite que o usuário libere o arquivo e envie o arquivo modificado para o servidor. Com isso, todos podem utilizar agora o arquivo para o modo escrita.

Verificando o aprendizado

Questão 1

Azure é uma plataforma de computação em nuvem operada pela Microsoft para gerenciamento de aplicativos via datacenters gerenciados e distribuídos globalmente pela Microsoft. Ele fornece software como serviço (SaaS), plataforma como serviço (PaaS) e infraestrutura como serviço (IaaS) e suporta muitas linguagens de programação, ferramentas e frameworks diferentes, incluindo software e sistemas específicos da Microsoft e de terceiros. Nesse contexto, analise as afirmativas a seguir.

I. O Azure Boards provê um conjunto de templates nativos predefinidos para suporte aos processos de gerenciamento do desenvolvimento baseado em Agile, Scrum e Kanban.

II. O Azure Artifacts é um orquestrador que permite a automação de tarefas que suporta a entrega contínua de software desde sua fase de compilação, passando por teste até sua publicação e execução.

III. A entrega contínua geralmente significa que as alterações feitas por uma equipe de desenvolvimento em uma aplicação são automaticamente testadas e carregadas em um repositório (como o GitHub ou um registro de aplicações em container), onde podem ser implantadas em um ambiente de produção em tempo real.

Assinale a alternativa correta cujas afirmativas se relacionam com computação em nuvem:

A

As afirmativas I, II e III.

B

Apenas as afirmativas II e III.

C

Apenas as afirmativas I e II.

D

Apenas as afirmativas I e III.

E

Apenas a afirmativas III.



A alternativa D está correta.

O Azure Pipelines é um serviço CI (integração contínua) e CD (entrega contínua) cheio de recursos. Ele funciona com seu provedor Git de preferência e pode ser implantado na maioria dos principais serviços de nuvem.

Questão 2

Qual sistema de controle de versão é centralizado e, geralmente, os membros da equipe têm apenas uma versão de cada arquivo em seus computadores de desenvolvimento, sendo os dados históricos mantidos somente no servidor?

A

Git

B

TFVC

C

Repos

D

Pipelines

E

Boards



A alternativa B está correta.

O TFVC é um repositório de código fonte e controlador de versão. Esse tipo de ferramenta é especialmente útil quando trabalhamos em equipe e precisamos ter a garantia de que o trabalho de um integrante da equipe não vá sobrepor o trabalho de outros.

Criação do projeto no Azure DevOps

Aprenda como criar um projeto no Azure DevOps neste vídeo, em que é abordada a criação de tarefas e movimentações nos cards do board.



Conteúdo interativo

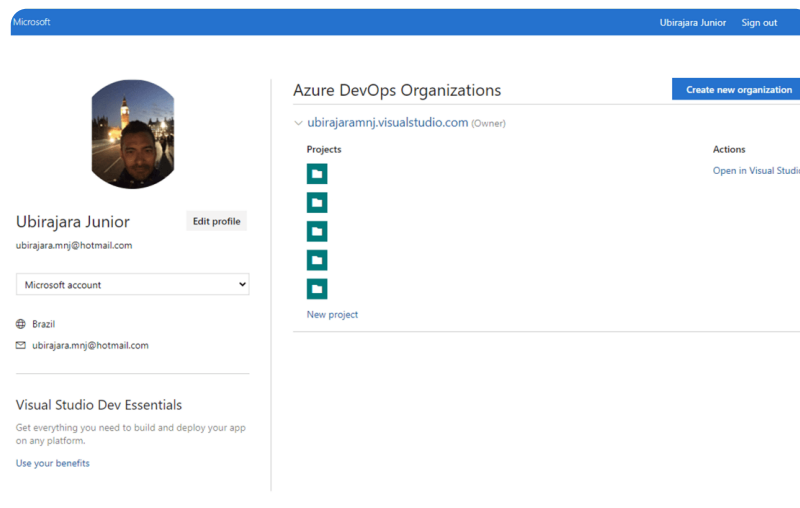
Acesse a versão digital para assistir ao vídeo.

Acessando o Azure DevOps

Agora que já conhecemos os principais conceitos relacionados com a plataforma Azure e Azure DevOps, vamos colocar a mão na massa!

Vamos acessar a plataforma Azure DevOps e conhecer as ferramentas que apresentamos até aqui. Faremos um passo a passo, desde o cadastro até termos nosso primeiro board funcionando, além da publicação de uma API no serviço de computação App Services da Azure, tudo de maneira integrada.

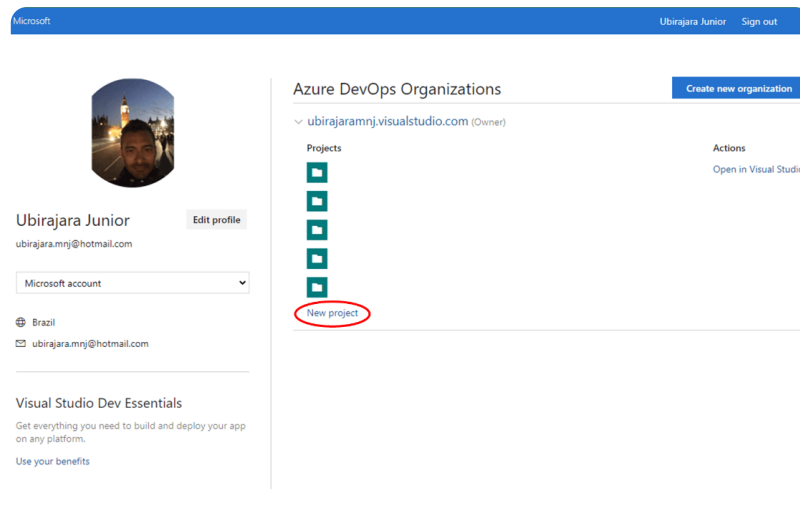
Para acessar o Azure DevOps, entre no site do visualstudio.com e faça login com sua conta Microsoft. Terminando o login você será direcionado a sua página principal, onde estarão listadas as organizações que sua conta está relacionada.



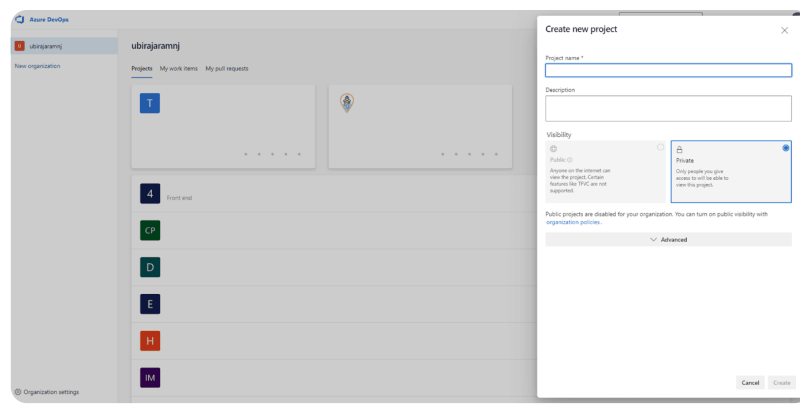
Criando o projeto

Após o login, na página de detalhes da conta, você tem a opção de criar uma organização, porém essa opção é destinada a quem vai administrar o Azure DevOps para a empresa como um todo.

No nosso caso, vamos criar o projeto DevOpsNaPratica dentro da organização de exemplo ubirajaramnj, para isso basta clicar no botão “New project” (Novo projeto) abaixo da lista de projetos preexistente (se houver).



Após clicar, uma nova página será aberta para a criação do projeto, nessa página você encontrará uma lista de projetos existentes (caso haja) e uma aba para que as informações do novo projeto sejam preenchidas.



Preencha os dados do nome do projeto, uma descrição e clique no botão “Advanced” (avançado). Na guia que se abrirá você deve configurar o sistema de controle de versão e o processo de itens de trabalho que é a abordagem de gerenciamento de projeto a ser utilizada, no nosso exemplo utilizaremos o Git como “Version control” (controle de versão) e “Basic” (básico) como abordagem de gerenciamento de projeto. Após as seleções, clique no botão “Create” (criar).

Create new project

Project name *

DevOps na Pratica

Description

Projeto de exemplo para o módulo DevOps na Prática.

Visibility

Public

Anyone on the internet can view the project. Certain features like TFVC are not supported.

Private

Only people you give access to will be able to view this project.

Public projects are disabled for your organization. You can turn on public visibility with [organization policies](#).

Advanced

Version control

Git

Work item process

Basic

Agile

Basic

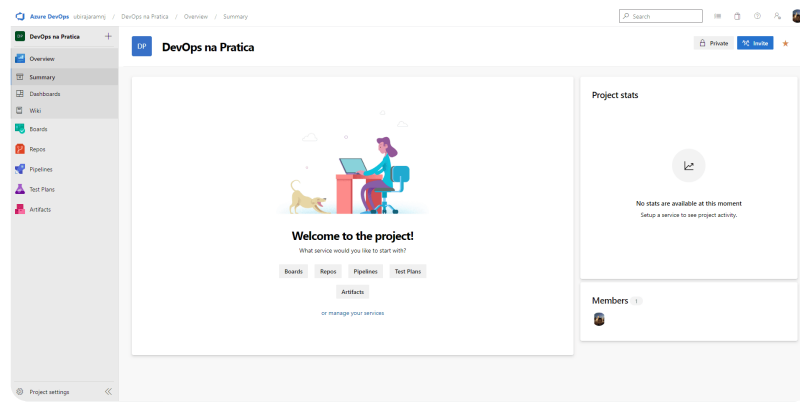
CMMI

Scrum

Cancel

Create

Após alguns segundos você será redirecionado para a página do projeto recém-criado. Pronto! Agora é hora de configurar o nosso projeto. No menu de opções, na lateral esquerda, você tem acesso aos principais serviços do Azure DevOps: o Boards, o Repos, o Pipelines, o Test Plans e o Artifacts, vamos a seguir configurar cada um deles.

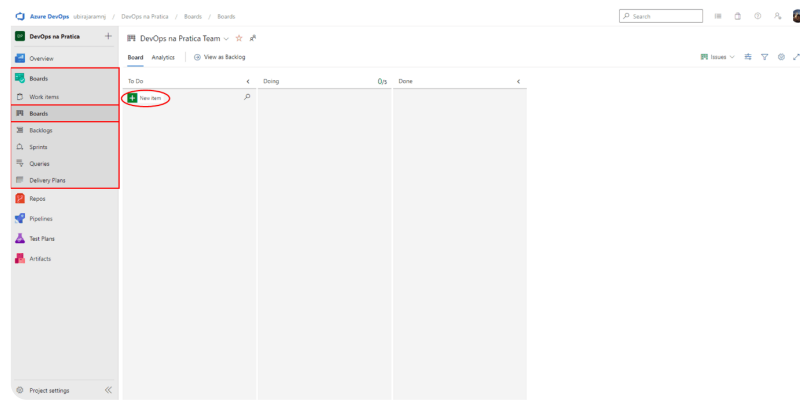


Gerenciamento do projeto

Criação de tarefas

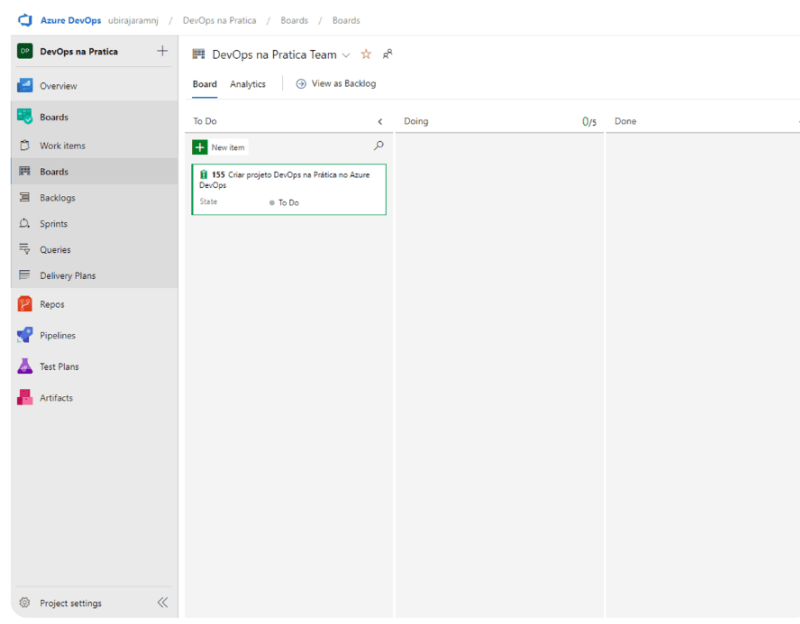
Conforme configuramos no momento da criação do nosso projeto, vamos utilizar o padrão “Basic” para o gerenciamento do nosso projeto de exemplo. Esse padrão é realmente básico, trata-se de um Kanban no qual as tarefas serão movidas entre os estados “To Do” (a fazer), “Doing” (em andamento) e “Done” (pronto), ou seja, uma tarefa vai passar por apenas três fases desde a sua criação até sua finalização. O Kanban é uma das abordagens ágeis mais utilizadas no mercado e o passo inicial para quem deseja iniciar uma cultura ágil em seus projetos.

Agora vamos adicionar as tarefas que vamos realizar ao longo do nosso estudo de caso. No menu da lateral esquerda, clique no botão “Boards” e em seguida na opção “Boards”:

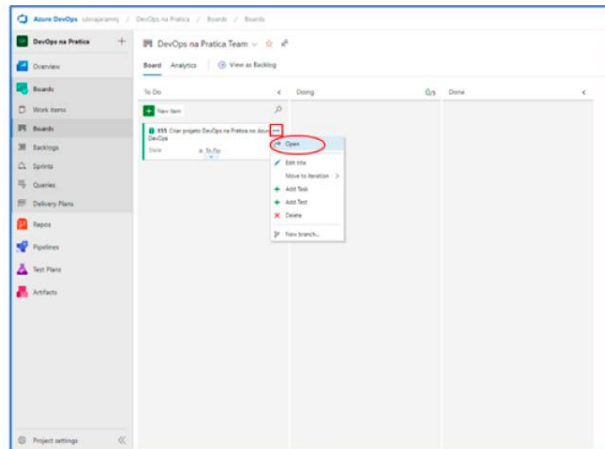


Nesta página temos três colunas que representam o que fazer com um “Work Item” (item de trabalho), podendo estar: a fazer pela “To Do”, em execução pela “Doing” e, encerrado, pela “Done”.

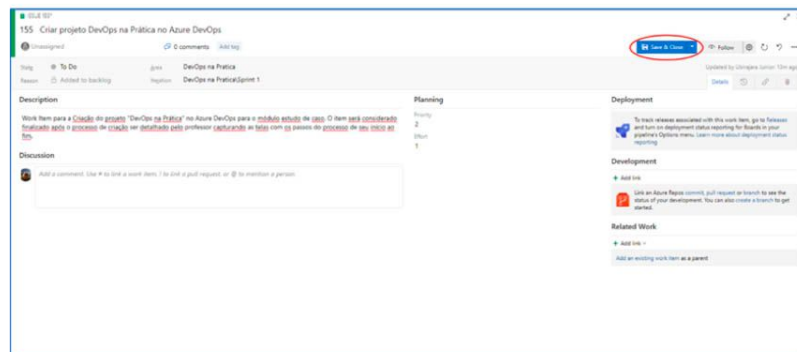
Para adicionar um novo item de trabalho, clique no botão “New Item” (novo item) e um “Card” (cartão) será adicionado a coluna “To Do”. Insira um texto que descreva o objetivo do item de trabalho e pressione enter, ao final, seu primeiro “Work Item” foi criado!



Muito simples, certo? Mas vamos explorar um pouco mais as características de um “Work Item”. Passando com o mouse sobre o “Card”, uma opção sinalizada com três pontos será exibida e, ao clicar nela, um menu de opções será exibido. Clique na opção “Open” (abrir).



Ao clicar na opção “Open”, os detalhes do card serão exibidos e você pode, e deve, preencher os detalhes daquela “Work Item”.



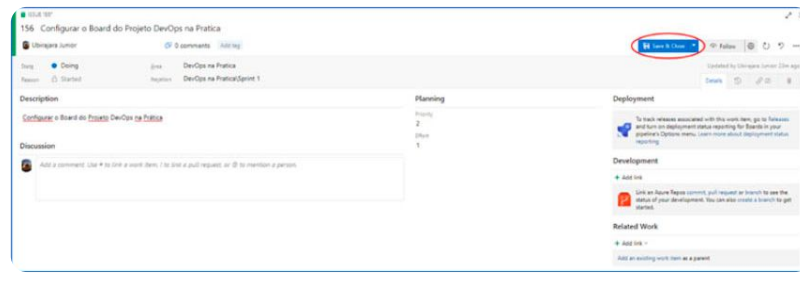
Na página de detalhes do “Work Item”, temos alguns campos extremamente relevantes como o campo “Unassigned” (não atribuído), esse campo informa quem será a pessoa responsável por executar esse item, e como ainda não atribuímos essa tarefa a alguém do time, ela segue como não atribuída. Em seguida, temos o campo “Area” (área), que tem por objetivo relacionar o item ao nosso projeto DevOps na Prática. O campo “Iteration” (iteração) relaciona o item ao momento em que ele deve ser executado (falaremos sobre esse campo em breve). No campo “Description” (descrição) informamos a descrição do item e preenchemos também outros campos importantes, como o “Priority” (prioridade), que recebe um valor de 1 a 4 e tem o objetivo de determinar qual o valor de prioridade desse item em relação aos seus pares. Também preenchemos o campo “Effort” (esforço), que determina o grau de esforço que o item requer para ser finalizado, aqui se determina uma métrica para essa avaliação, no nosso projeto de exemplo vamos usar horas, e esse item requer um esforço equivalente a uma hora. Por fim, temos o campo “Discussion” (discussão), que é um campo onde os membros do time podem trocar informações sobre o item a ser desenvolvido.



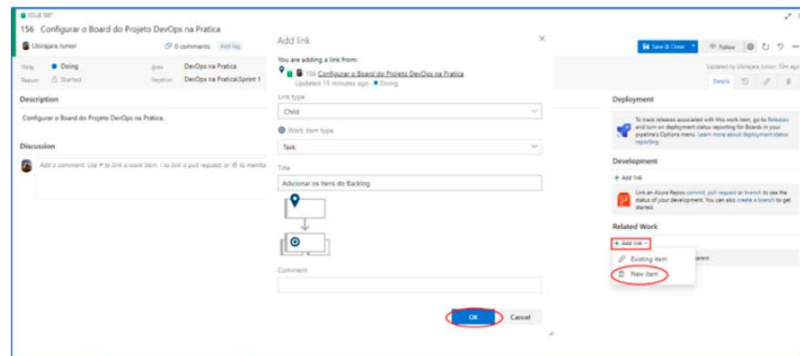
Saiba mais

Temos alguns campos relacionados ao “Deployment” (publicação), utilizado quando o item a ser executado gera um artefato de software a ser empacotado e publicado para o usuário identificar qual versão de publicação aquele item foi entregue. Também podemos adicionar o link para o “Pull request” (requisição de integração) do código gerado pela execução do item.

Vamos agora criar a tarefa “Configurar o board do projeto DevOps na Prática”, de acordo com a imagem a seguir. Caso não existam tarefas menores, basta clicar em “Save & Close” (salvar e fechar).

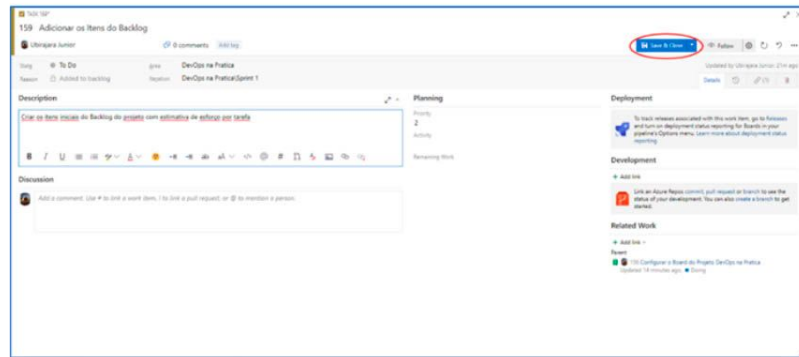


Caso você identifique que o item de trabalho possui tarefas menores que devem ser explicitadas e executadas para a finalização do item, essas tarefas devem ser criadas e associadas ao item de trabalho. Para isso, clique no botão “Add link” (adicionar associação), o menu apresentará duas opções “Existing item” (item existente) e “New Item” (novo item), permitindo que a associação ocorra em itens pré-existentes ou criar item, vamos clicar em “New item” e na janela “Add link” preencha o título da tarefa e clique em “OK”.



Agora vamos então criar as tarefas via janela “Add link” na tarefa “Configurar o board do Projeto DevOps na Prática”, sendo direcionada a tarefa “Adicionar os itens do Backlog”.

Uma janela para preencher os detalhes da tarefa será aberta e, ao final, basta clicar em “Save & Close” (salvar e fechar).

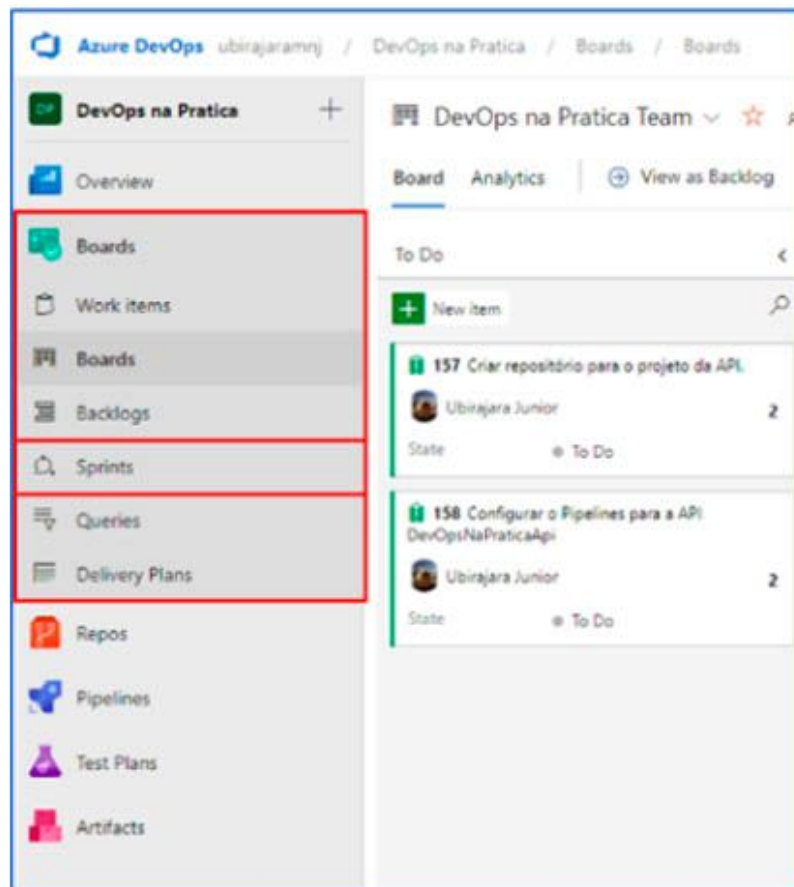


Repita os passos de criação de tarefas para cada novo item de trabalho que você precise no seu projeto, vejamos:

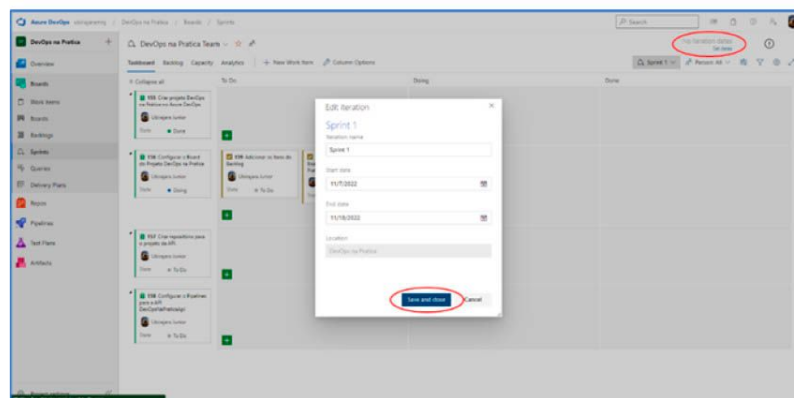
- Tarefa: “Criar repositório para o projeto da API”.
- Tarefa: “Configurar o Pipeline para a API DevOpsnaPraticaApi”.

Datas de início e término da sprint

Agora que temos todos os itens de trabalho e tarefas do backlog cadastrados, vamos à última configuração do Boards do Azure DevOps que é responsável pelas iterações, ou seja, as datas de início e fim de cada sprint. Para isso, clique no botão “Sprints” dentro da seção “Boards”.

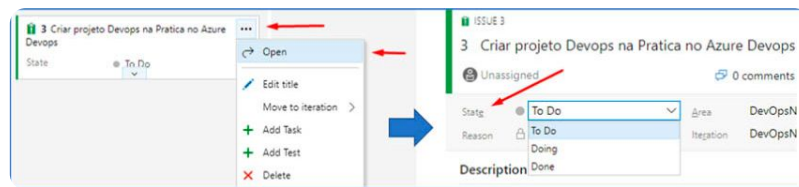


Na página “Sprints”, no canto superior direito, clique no link “Set dates” (definir datas) para que a página de edição da iteração seja apresentada e preencha o nome da iteração (no exemplo vamos manter Sprint 1). Defina a “Start date” (data inicial) e a “End date” (data final) e por fim, clique no botão “Save and close” (salvar e fechar). Pronto, as datas limites da iteração estão definidas, ou seja, é esperado que todos os itens de trabalho que estiverem relacionados à iteração “Sprint 1” sejam finalizados entre as datas 07/11/2022 e 18/11/2022, conforme exemplo.



Movimentações nos cards do Board

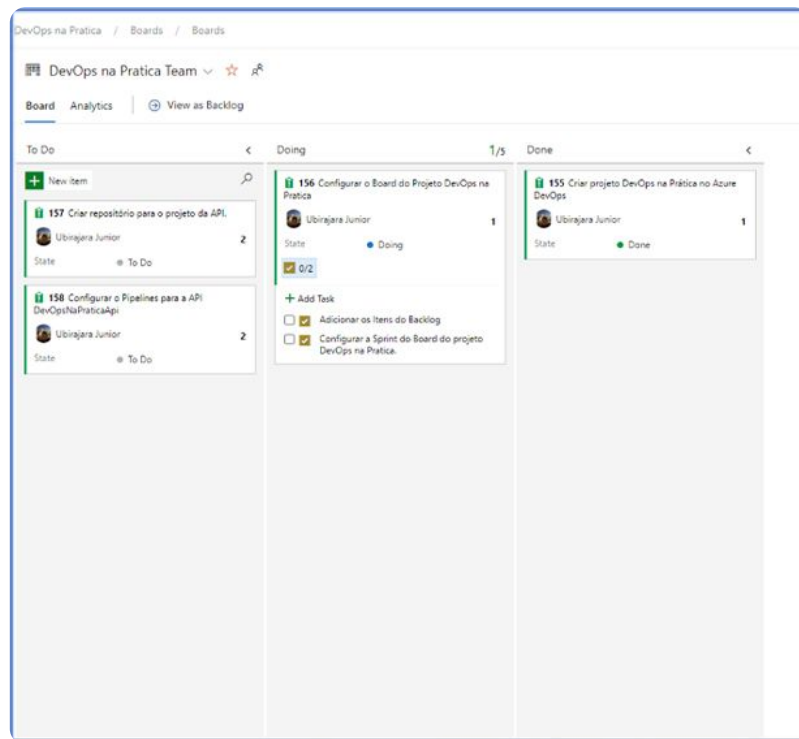
Essas movimentações podem ser realizadas por meio da alteração do “state” (estado) da tarefa selecionada, de acordo com esta imagem.



Vamos agora simular as seguintes movimentações nos cards do board, alterando os respectivos estados (states):

- Tarefa: “Criar projeto Devops na Prática no Azure Devops” para “Done” (id 155 na imagem).
- Tarefa: “Configurar o Board do Projeto DevOps na Prática” para “Doing” (id 156 na imagem).
- Tarefa: “Criar repositório para o projeto da API” para “To do” (id 157 na imagem).
- Tarefa: “Configurar o Pipeline para a API DevOpsnaPraticaApi”, status “To do” (id 158 na imagem).

Após essas alterações do state, teremos a seguinte configuração no Board.



Configurando o Azure Repos

Confira os principais aspectos das configurações no Azure.

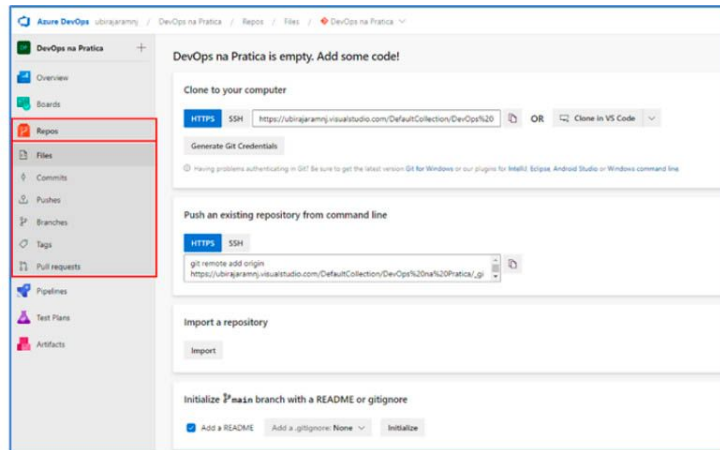


Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Criação de repositório no Azure

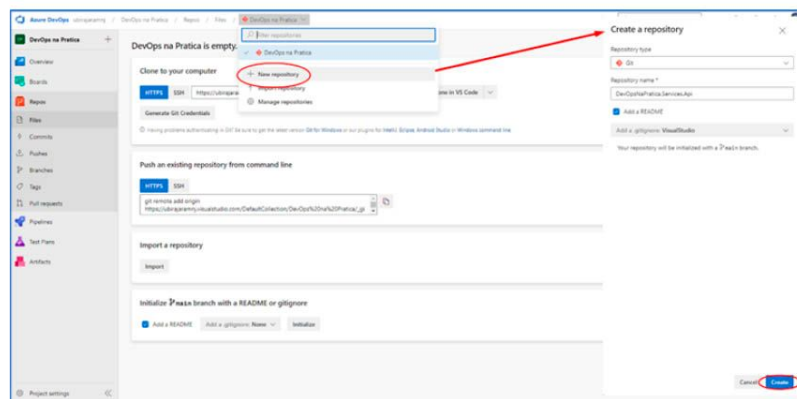
Agora que nosso board está corretamente configurado e com os itens de trabalhos definidos, vamos configurar o Azure Repos. Para isso, clique no ícone Repos no menu lateral esquerdo para acessar a página dos repositórios do projeto. Todo projeto criado no Azure DevOps possui repositório de código com o nome do projeto criado, no nosso caso, temos já criado automaticamente o repositório “DevOps na Prática”.



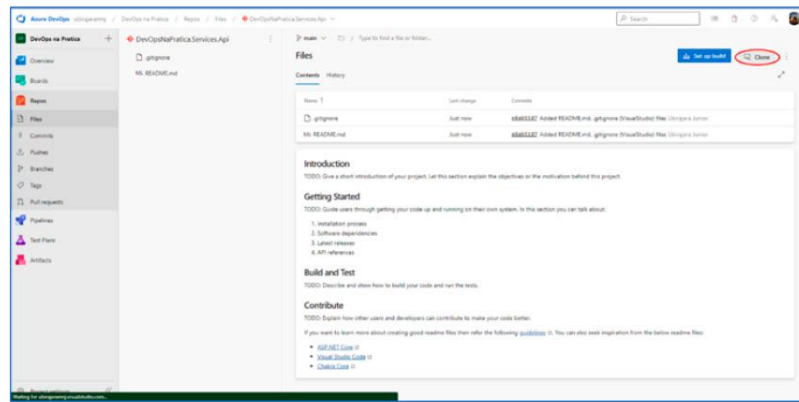
Este repositório está pronto para uso e para facilitar ainda mais nosso trabalho, o Azure DevOps apresenta as linhas de comando para você começar a usar o repositório, permitindo você criar um “Clone” (criar uma cópia local), fazer um “Push” (enviar um repositório local para o Azure Repos) ou “Import” (importar um repositório remoto de outro serviço git).

Criação de repositório local

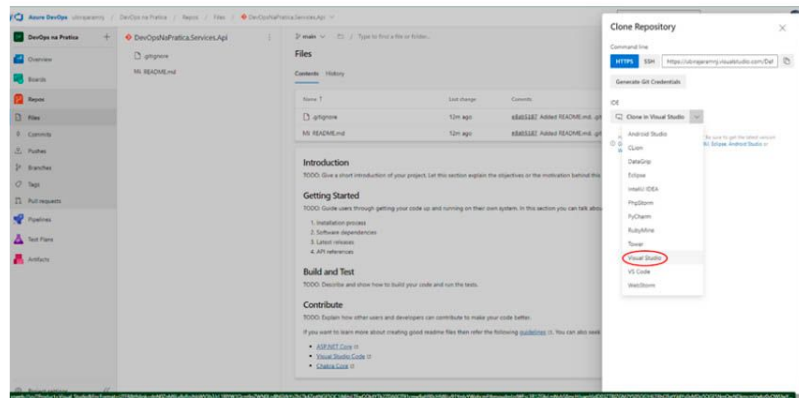
Para enriquecer nosso exemplo, vamos criar um repositório para nossa “DevOps na Prática Api” e clonar utilizando o Visual Studio. Para isso, basta clicar no nome do repositório na barra superior da página e em seguida no link “New repository” (novo repositório). Na guia “Create a repository” (Criar repositório), preencha com o nome do repositório, e selecione o arquivo .gitignore que melhor se adeque ao seu tipo de projeto, nesse caso utilizaremos o padrão Visual Studio para projetos desenvolvidos utilizando a linguagem C#.



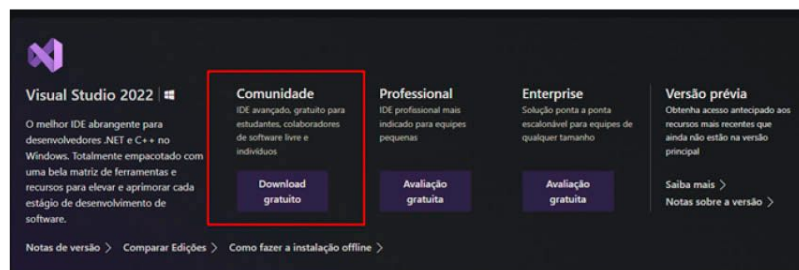
Por fim, basta clicar no botão “Create” (criar) que o repositório será criado pelo Azure DevOps. No final do processo, teremos a página de detalhes do repositório, onde faremos o “Clone” local do repositório, ou seja, uma cópia local em nosso dispositivo.



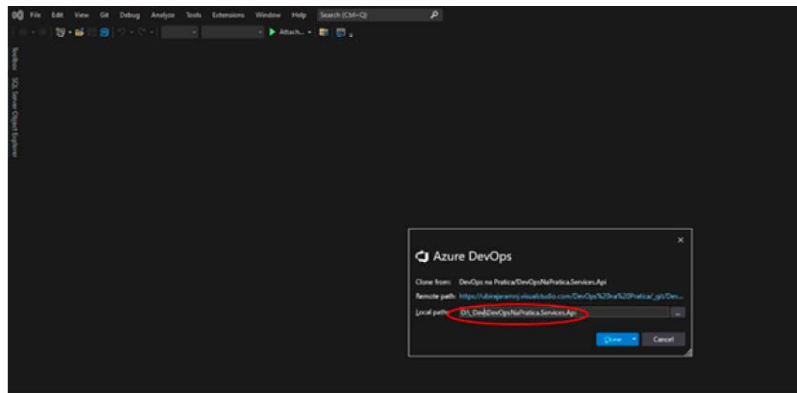
Para fazer um clone no local do repositório do Azure DevOps, basta clicar no botão “Clone” no canto superior esquerdo onde as opções de cópia serão exibidas. Você pode fazer a cópia usando a linha de comando ou utilizando uma das várias IDEs que estão pré-configuradas para a cópia, neste exemplo vamos utilizar o Visual Studio, clicando no botão da lista.



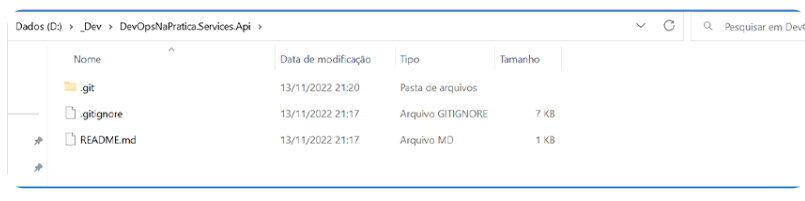
Caso não tenha o Visual Studio instalado, acesse o site do Visual Studio e faça o download da versão de comunidade, como ilustra a imagem.



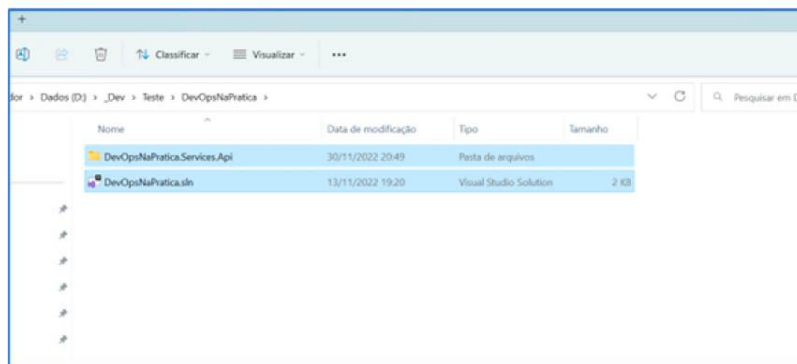
Haverá um redirecionamento para o aplicativo do Visual Studio, no qual temos que informar ao diretório o local onde será realizada a cópia do repositório e em seguida clicar no botão “Clone”.



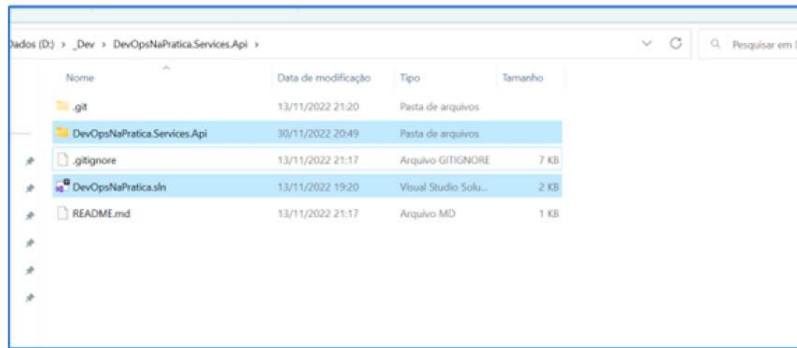
Abra o Windows Explorer e navegue até a pasta que você acabou de clonar, ela terá a seguinte estrutura:



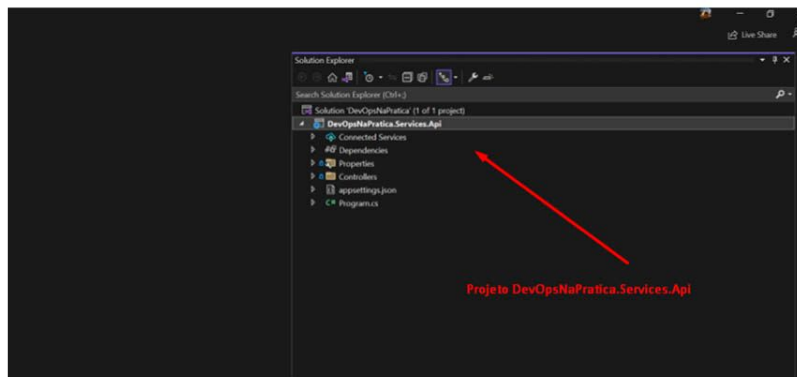
Agora baixe o arquivo [DevOpsNaPratica.zip](#) e descompacte em uma pasta qualquer, navegue no conteúdo da pasta descompactada e selecione os seguintes arquivos destacados abaixo:



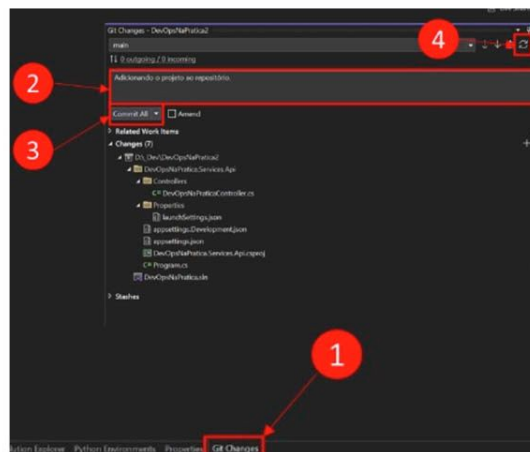
Copie e cole os arquivos dentro da pasta da qual você realizou o clone do repositório. A estrutura da pasta ficará desta forma:



Dê um duplo clique no arquivo DevOpsNaPratica.sln e o Visual Studio abrirá a seguinte solução:



Selecione a aba "GitChanges" (1) ou "Alterações do Git" e em seguida insira um texto no campo de "Descrição do Commit" (2). Depois clique no botão "Commit All" (3) e, por fim, clique no botão "Sync (Pull and Push)" (4):



Pronto! Neste ponto temos o Azure Repos configurado, com uma cópia local em nosso dispositivo e o projeto no nosso repositório.

Implementando um Azure Pipelines

Saiba mais nesse vídeo sobre como implementar um Azure Pipelines, incluindo criação, configuração e execução do Pipelines.



Conteúdo interativo

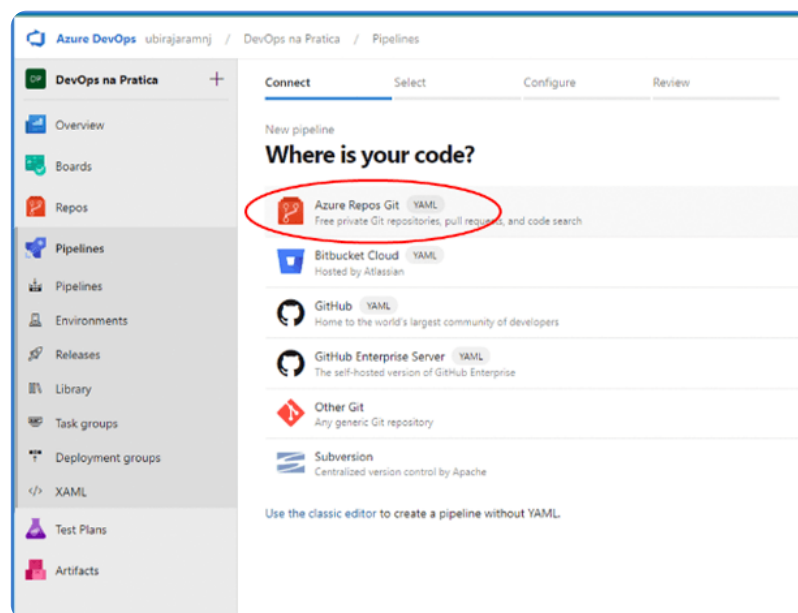
Acesse a versão digital para assistir ao vídeo.

Criação do Pipelines

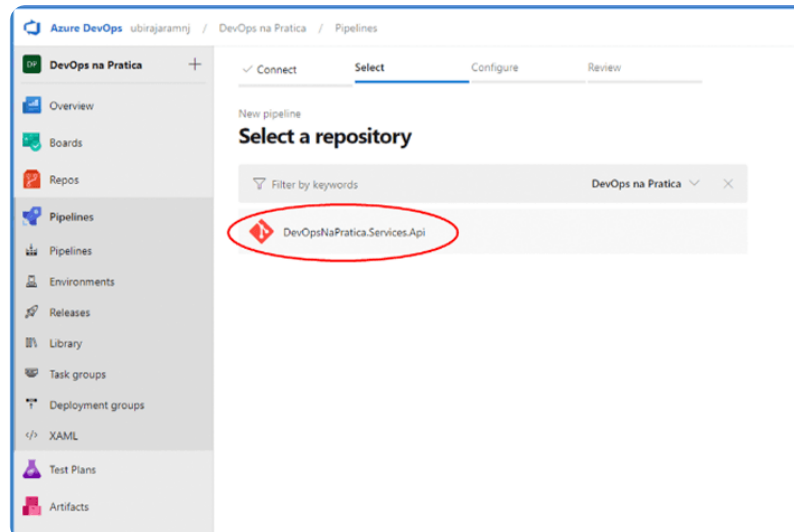
Azure Board configurado, bem como Azure Repos, agora vamos configurar o Azure Pipelines, que será responsável por efetuar o processo de integração contínua. Tal processo consiste em efetuar o “Build” (compilação) da solução, executar os “Unit Tests” (testes de unidade) e geração do “Package” (pacote) para o “Deploy” (publicação) no nosso ambiente de teste. Assim como nos serviços anteriores, para acessar a página de configuração do Azure Pipelines, basta clicar no ícone “Pipelines” no menu esquerdo e em seguida clicar no botão “Create Pipeline” (criar pipeline).



Em seguida somos direcionados para a página de configuração do Pipelines com o conjunto de passos necessários para configuração. O primeiro deles é a definição do repositório de código que está na “DevOps na Prática Api”, que acabamos de criar na última seção. Clique em “Azure Repos Git”.



Em seguida, selecione o repositório que acabamos de criar: DevOpsNaPratica.Services.Api.



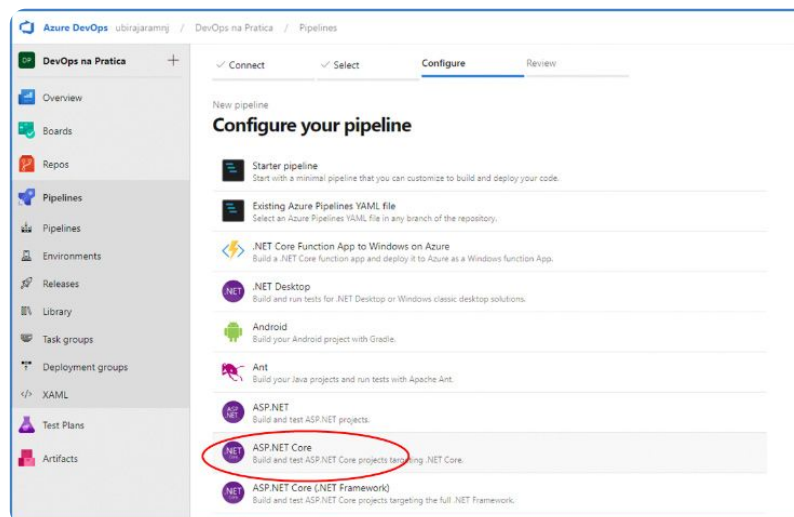
Captura de tela da criação de um pipeline.

Configuração do Pipelines

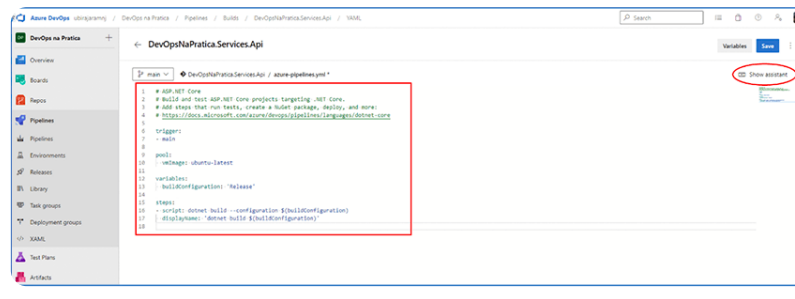
Agora que estamos com o repositório selecionados, vamos dar início à configuração do nosso pipeline. O pipeline que vamos construir terá três passos:

- 1 – Build (compilação) – Responsável pela compilação do projeto.
- 2 – Publish (empacotamento) – Responsável por empacotar o artefato de software gerado.
- 3 – Release (publicação) – Responsável por publicar o artefato gerado no serviço Azure App Service, onde será executado e ficará acessível aos usuários.

Na tela “Configure your Pipeline” (configurar seu pipeline) clique no botão “Show more” (mostrar mais) e selecione a opção ASP.NET Core.



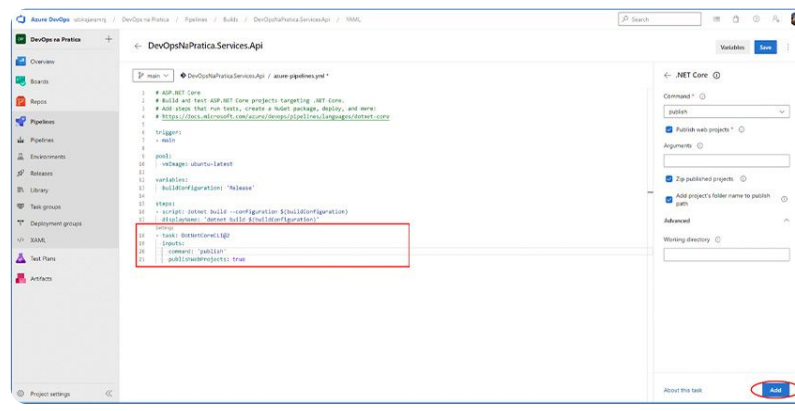
Nesse momento o arquivo de configuração do Pipelines (YAML) padrão para ASP.NET Core é apresentado e já contém o primeiro passo do nosso pipeline, que é o “Build”.



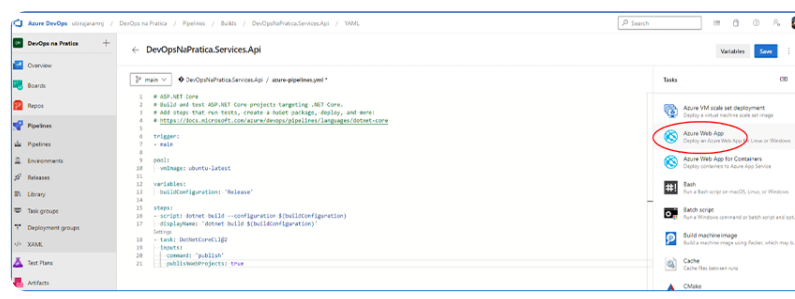
Agora vamos adicionar o segundo passo que é o “Publish” (empacotamento), para isso clique no link “Show assistant” (mostrar assistente) e selecione a opção “.NET Core”.



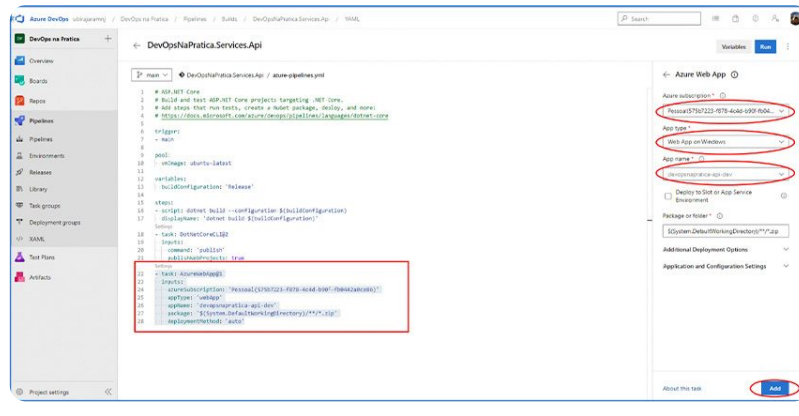
A guia do .NET Core será exibida, simplesmente clique no botão “Add” (adicionar) e o arquivo de configuração do Pipelines (YAML) será complementado com os códigos necessários para que esse passo seja executado.



E para finalizar, vamos configurar o último passo que é o “Release” (publicação): ainda no assistente de tarefas, role a barra até encontrar a opção “Azure Web App”.

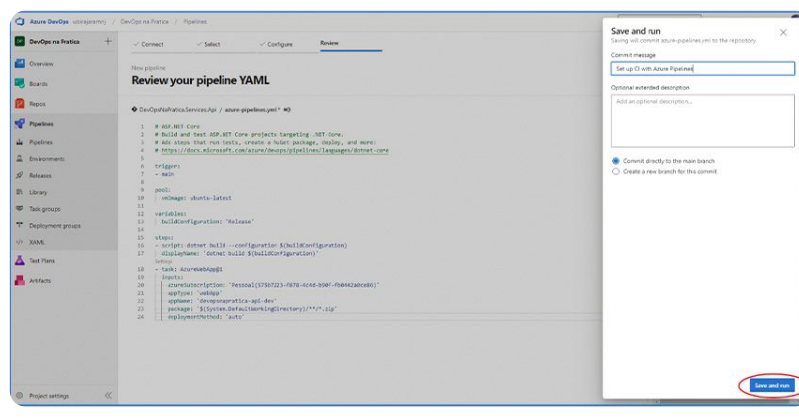


Ao clicar no ícone, a página de configuração para a publicação será exibida, preencha os campos “Azure subscription” (subscrição da Azure), “App type” (tipo de aplicação) e “App name” (nome da aplicação), onde o artefato deve ser publicado e, por fim, clique no botão “Add” (adicionar). Assim como no passo anterior, o arquivo de configuração do Pipelines (YAML) será configurado com a configuração para publicação.

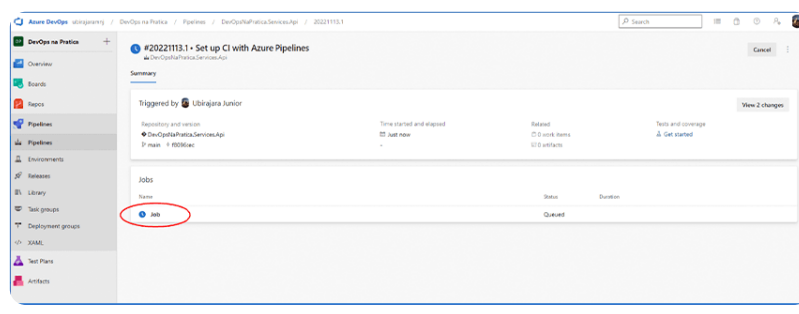


Execução do Pipelines

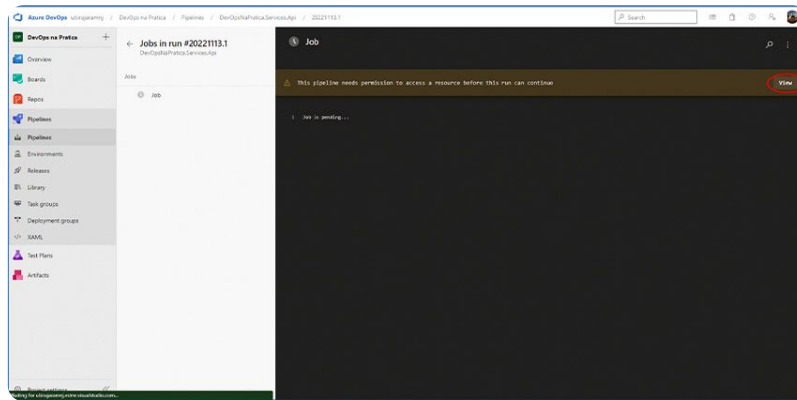
Pronto! Agora que temos todos os passos configurados, clique no botão “Save and run” (salvar e executar). Um commit para o repositório será realizado adicionando o arquivo “azure-pipeline.yml” que acabamos de criar e a tentativa de execução será iniciada.



Durante a tentativa da primeira execução, o “Job” (executor) ficará travado, clique na linha do “Job” para ter acesso aos detalhes de execução.

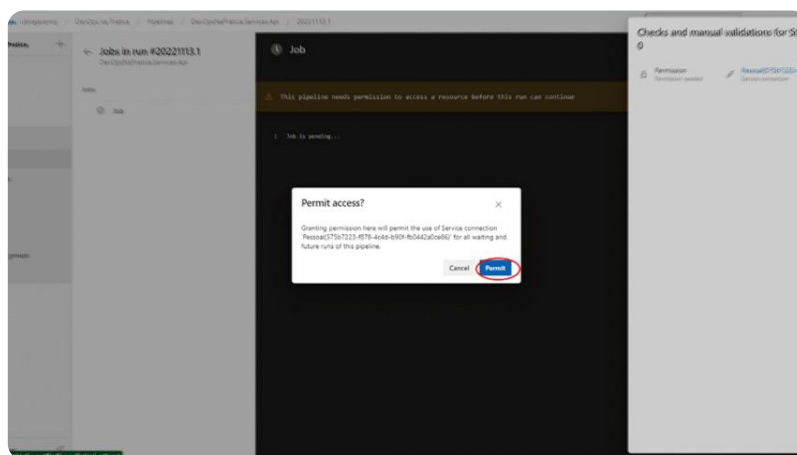


Ao analisar os detalhes de execução, haverá uma mensagem informando que o pipeline precisa de uma permissão de acesso ao recurso para poder continuar, apenas clique em “View” (visualizar).

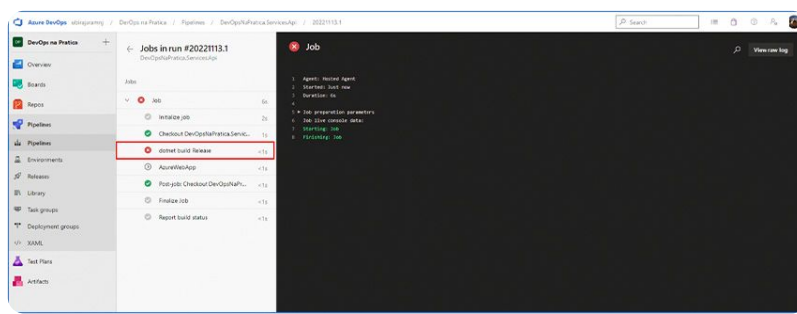


Captura de tela da criação de um pipeline.

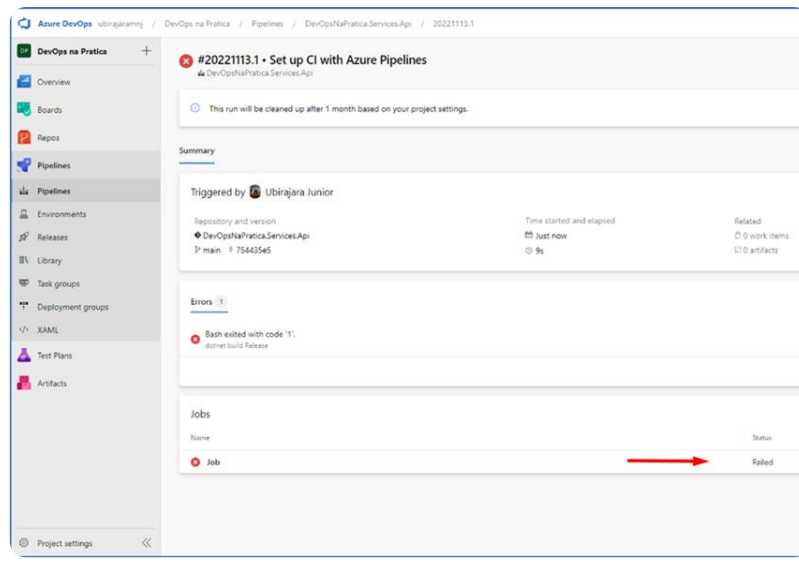
Em seguida, na guia “Checks and manual validations for Stage” clique no botão “Permit” (permitir) e em seguida no botão “Permit” na janela que irá surgir.



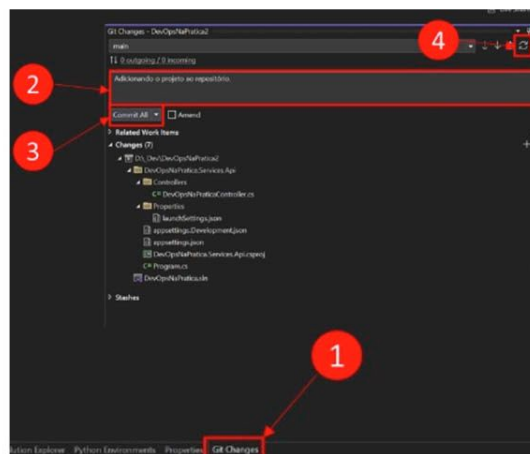
O “Job” iniciará sua execução, mas logo falhará. Isso se deve à falta do projeto no repositório, até agora só temos o repositório com o arquivo de configuração, e para vermos o pipeline ter sucesso, precisamos submeter o projeto da API para ele.



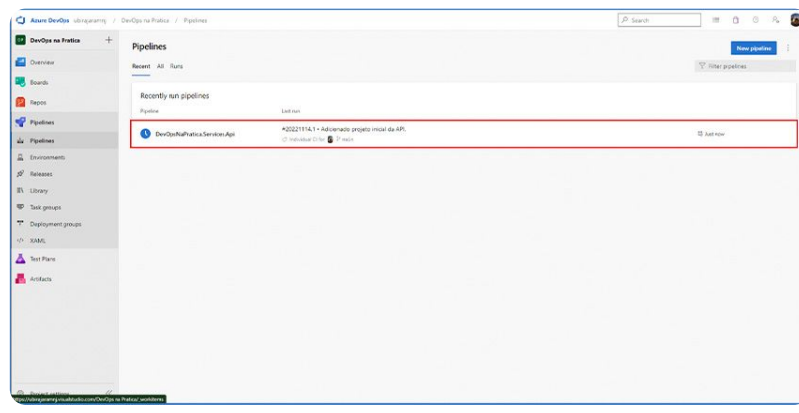
Na tela principal do Pipelines, podemos identificar com facilidade a falha na execução (status do Job = “).



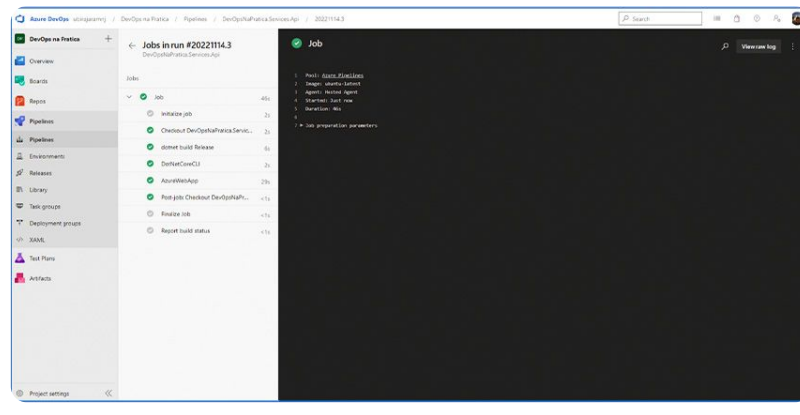
Agora o que precisamos é submeter o projeto para o repositório. Para isso, dentro do Visual Studio, com o projeto devidamente dentro da pasta copiada do repositório, basta acessarmos a guia “Git changes” (1), visualizar os arquivos adicionados ao repositório, relacionar o item de trabalho com essa alteração, escrever uma mensagem, informando quais foram as alteração (2), e então efetuar o “commit” (3). Por fim, basta efetuar um “Pull” (empurrar) e (4) as alterações locais para o Azure Repos.



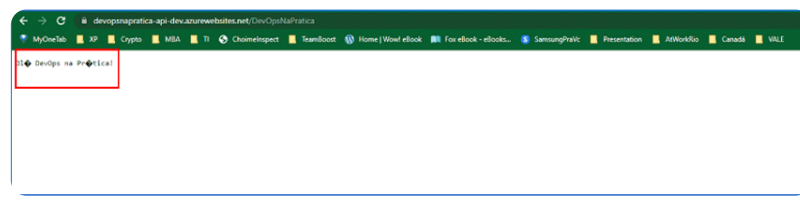
Assim que o repositório do Azure DevOps receber as alterações, o Pipelines será informado de que novas alterações foram adicionadas à base de código.



Assim, será iniciada a execução do conjunto de passos configurados no Pipelines, o “Build” (versão compilada de um programa.), o “Publish” (Publicar) e o “Release” (entrega de versão), o que acarretará a publicação da API no serviço Azure Web App.

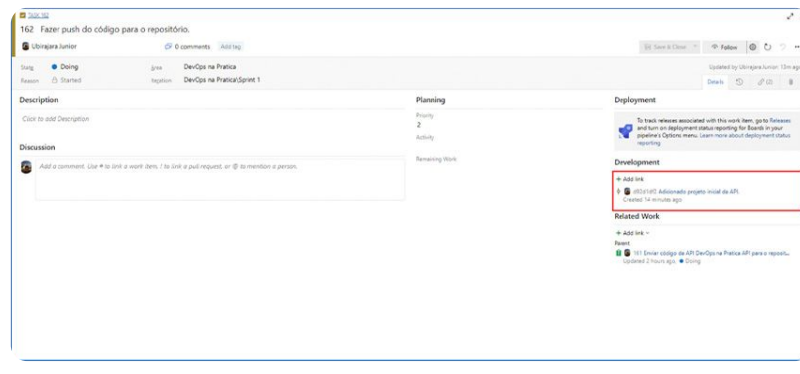


Após todos os passos terem sido executados com sucesso, podemos acessar o link do serviço da Azure Web App e, então, teremos a mensagem recebida pela API.



Pronto! Tudo funcionando conforme esperado, e o principal, de maneira automatizada!

Para finalizar, você lembra que no passo de enviar o código da API do local para o Azure Repos, nós informamos qual era o item de trabalho relacionado àquela alteração? Pois é, essa alteração fica também relacionada lá no Azure Boards, veja na imagem.



Agora temos todo o nosso Azure DevOps configurado, desde o Board até os Pipelines, garantindo gestão, rastreamento e automação do desenvolvimento do projeto como um todo.

Verificando o aprendizado

Questão 1

Qual metodologia propõe o uso de cartões para indicar e acompanhar o andamento da produção, tratando-se de um sistema visual que busca gerenciar o trabalho conforme ele se move pelo processo?

A

Kanban

B

Scrum

C

Board

D

Extreme Programming - XP

E

Sprints



A alternativa A está correta.

Kanban é um método ágil que se baseia em técnicas que controlam e fazem a gestão dos fluxos de produção para as empresas.

Questão 2

Qual tecnologia é um conjunto de ferramentas de controle de versão que você pode usar para gerenciar seu código. Se o projeto de software for grande ou pequeno, o uso do controle de versão assim que possível será uma boa ideia?

A

Azure Artifacts

B

Azure Test Plan

C

Azure Pipelines

D

Azure Repos

E

Azure Boards



A alternativa D está correta.

O Azure Repos fornece dois tipos de controle de versão: Git, controle de versão distribuída; e controle de versão do Team Foundation (TFVC), controle de versão centralizado.

Considerações finais

Apresentamos a utilização de uma abordagem DevOps que pode ajudar você a tornar o seu projeto de software mais produtivo, permitindo a medição dessa produtividade, bem como melhor qualidade do resultado dessa produtividade, ou seja, a sua entrega.

Vimos como esse conjunto de práticas e técnicas adotadas pela abordagem DevOps está revolucionando a maneira como as empresas se organizam com o objetivo de entregar inovação e valor para seus clientes, de forma rápida, com alta qualidade e frequente.

Enfim, a metodologia DevOps é uma abordagem de cultura, automação e design de plataforma que tem como objetivo agregar mais valor aos negócios e aumentar sua capacidade de resposta às mudanças por meio de entregas rápidas de serviços e de alta qualidade.

Podcast

Para encerrar, ouça mais sobre os modelos de infraestrutura e as possibilidades do Azure.



Conteúdo interativo

Acesse a versão digital para ouvir o áudio.

Explore +

Confira as indicações que separamos especialmente para você!

Pesquise sobre:

- O que é o Azure DevOps?
- Azure DevOps e integração de compilação baseada em dependência da IBM.
- AWS Toolkit for Azure DevOps.
- Serviço Oracle Cloud Infrastructure DevOps.

Referências

AZURE MICROSOFT. **Azure Artifacts**. Consultado na internet em: 15 de dez. 2022.

AZURE MICROSOFT. **Azure Boards**. Consultado na internet em: 15 de dez. 2022.

AZURE MICROSOFT. **Azure Pipelines**. Consultado na internet em: 15 de dez. 2022.

AZURE MICROSOFT. **Azure Repos**. Consultado na internet em: 15 de dez. 2022.

AZURE MICROSOFT. **Azure Test Plans**. Consultado na internet em: 15 de dez. 2022.

AZURE MICROSOFT. **Microsoft Azure**. Consultado na internet em: 15 de dez. 2022.