

製品オプションを理解する

100 XP

4 分

ソフトウェア開発者や運用担当者は、組織のニーズを満たす実用的なソフトウェア システムを作成しようと努めています。ただし、短期的な目標が相互に食い違った結果、技術的な問題、遅延、ダウンタイムが発生することがあります。

DevOps は、技術的なチームを共通の目標に向けて取り組ませる新しいアプローチです。この整合を実現するため、組織では、ソフトウェア システムの継続的な開発、保守、デプロイを自動化するプラクティスとプロセスが採用されています。その目的は、ソフトウェア変更のリリースまでの時間を短縮し、システムの継続的なデプロイ可能性を確保し、すべての変更が高品質の基準を満たすようにすることです。

DevOps のプラクティスおよびプロセスは、正常に行われれば、計画、プロジェクト管理、ソフトウェア開発者のコラボレーションのほか、運用チームや品質保証チームとのコラボレーションを含むソフトウェア開発のライフサイクルは言うまでもなく、会社のほぼすべての側面に影響します。ツールは、ほとんどのプラクティスとプロセスを自動化して適用するので、回避することは困難で不要になっています。

DevOps では、トップダウンで基本的な考え方を変更することが必要です。組織では、ソフトウェア ツールをインストールしたり、サービスを導入したりするだけでは、DevOps によって約束されるすべての特典を得ることはできません。

このモジュールでは、DevOps の一部の目標を達成するのに役立つ Microsoft のツールにのみ焦点を当てます。または、DevOps の力を完全に受け入れる準備ができていない組織の場合、クラウド開発アクティビティで技術チームをサポートできます。DevOps 全体の詳細について知りたい場合は、Microsoft Learn に、役立つ複数のラーニング パスとモジュールが用意されています。

Microsoft からは、ソース コード管理、継続的インテグレーションと継続的デリバリー (CI/CD)、テスト環境の作成の自動化を可能にするツールが提供されています。これらのツールの機能が重複しているように見えることがあります。そのため、このモジュールでは、いくつかの製品オプションについて説明し、ある製品を別の製品に優先して選択する場合について説明します。

製品オプション

大まかに言えば、3 つの主要なオフリングがあり、それぞれが特定の対象ユーザーとユース ケースを対象としており、さまざまなツール、サービス、プログラマティック API などが提供されています。

Azure DevOps Services

Azure DevOps Services は、ソフトウェア開発ライフサイクルのすべての段階に対応する一連のサービスです。

- **Azure Repos** は、ソフトウェア開発、DevOps エンジニアリング、ドキュメントの専門家が、レビューやコラボレーションのためにコードを公開できる一元的なソース コード リポジトリです。
- **Azure Boards** は、アジャイルなプロジェクト管理スイートであり、かんばんボード、レポート、アイデアの追跡、高度なエピックから作業項目や問題に至る作業などが含まれます。
- **Azure Pipelines** は、CI/CD パイプライン オートメーション ツールです。
- **Azure Artifacts** は、テストまたは配置パイプライン ステップに渡すことができる成果物 (コンパイルされたソース コードなど) をホスティングするためのリポジトリです。
- **Azure Test Plans** は、ソフトウェア リリースの前に品質を確保するために CI/CD パイプラインで利用できる自動テスト ツールです。

Azure DevOps は、オンプレミス サーバー ソフトウェアとして始まり、Microsoft のサービスとしてのソフトウェア (SaaS) オファリングに発展した大規模な機能セットを備えた成熟したツールです。

GitHub と GitHub Actions

GitHub は、オープンソース ソフトウェアの世界でおそらく最も人気のあるコード リポジトリです。Git は、分散型のソース コード管理ツールですが、GitHub は、プライマリ リモートとして機能する、ホストド バージョンの Git です。GitHub は Git 上に構築され、作業の調整、問題の報告とディスカッション、ドキュメントの提供などの関連サービスを提供します。以下の機能が提供されます。

- 共有ソース コード リポジトリ。ソース コードをメイン コード ベースにマージする前に、開発者がソース コードの Web ビューでコメントや質問を追加してコード レビューを実行できるようにするツールが含まれます。
- かんばんボードなど、プロジェクト管理が容易になります。
- 問題の報告、ディスカッション、追跡がサポートされます。
- CI/CD パイプライン オートメーション ツールがあります。
- 共同作業のドキュメント用の Wiki が含まれます。
- クラウドまたはオンプレミスから実行できます

このモジュールに最も関連のある GitHub Actions を使用すると、多くのライフサイクル イベントに対するトリガーでワークフローを自動化できます。そのような例の 1 つは、CI/CD "ツールチェーン" の自動化です。

ツールチェーンは、システム開発のライフ サイクル全体を通じて、ソフトウェア アプリケーションのデリバリー、開発、および管理で役立つソフトウェア ツールの組み合わせです。ツールチェーンの 1 つのツールの出力は、ツールチェーンの次のツールの入力になります。典型的なツール機能は、自動化された依存関係の更新の実行から、ソフトウェアの構築と構成、さまざまな場所へのビルド成果物の配信、テストなどまで多岐にわたります。

GitHub と Azure DevOps の多くの機能は類似しているため、組織でどの製品を選べばよいか迷うかもしれません。残念ながら、答えは簡単ではありません。

Azure DevOps と GitHub はどちらもパブリックとプライベートのコード リポジトリを使用できますが、GitHub にはパブリック リポジトリの長期にわたる歴史があり、数万のオープンソース プロジェクト所有者から信頼されています。GitHub は Azure DevOps より軽量なツールであり、オープンソース コードに寄与している個人の開発者に重点が置かれています。一方、Azure DevOps は、より重量のプロジェクト管理および計画ツール、きめ細かいアクセス制御を備えたエンタープライズ開発にいったんその重点が置かれています。

注意

選択肢は、Azure DevOps Services か、GitHub と GitHub Actions のどちらかに制限されているわけではありません。実際には、必要に応じてこれらのサービスを混在させ組み合わせることができます。たとえば、作業項目を追跡するために GitHub リポジトリを Azure Boards と一緒に使用できます。

Azure DevTest Labs

Azure DevTest Labs では、ソフトウェア プロジェクトのビルドを含む仮想マシン (VM) の構築、設定、破棄のプロセスを管理する自動化された手段が提供されます。これにより、開発者やテスト担当者は、さまざまな環境とビルドについてテストを実行できるようになります。また、この機能は VM に限定されません。ARM テンプレートで Azure にデプロイできるものは、DevTest Labs からプロビジョニングできます。必要な構成やツールが既にインストールされている事前作成されたラボ環境をプロビジョニングすれば、品質保証の専門家と開発者は大幅に時間を短縮できます。

古いバージョンのオペレーティング システムで新しい機能をテストする必要があるとします。Azure DevTest Labs は、要求に応じて自動的にすべてのものを設定できます。テストが完了すると、DevTest Labs によって VM がシャットダウンされてプロビジョニング解除されるので、使用していないときにコストを節約できます。コストを制御するため、管理チームは、作成できるラボの数、実行時間などを制限できます。