

どのようなときに Azure App Service を使田するか

100 XP

3 分

Tailwind Traders に関する調査では、アプリケーションを仮想化できる 2 つの方法を見てきました。別の方法として、アプリケーションのフロントエンド Web サイトを Azure App Service にデプロイする方法もあり、このようにするとアプリケーションの需要に簡単に対応できます。

App Service を使用すると、インフラストラクチャを管理することなく、任意のプログラミング言語で Web アプリ、バックグラウンド ジョブ、モバイル バックエンド、RESTful API を構築し、ホストできます。これにより、自動スケーリングと高可用性が提供されます。App Service によって、Windows と Linux がサポートされます。さらに、GitHub、Azure DevOps、または任意の Git リポジトリからの自動デプロイが可能になり、継続的デプロイ モデルに対応します。



このサービスとしてのプラットフォーム (PaaS) 環境を使用すると、Web アプリケーションの実行とスケーリングのためのインフラストラクチャは Azure で処理されるため、ユーザーは Web サイトと API ロジックに集中できます。

Azure App Service のコスト

アプリによって使用される Azure コンピューティング リソースに対して料金を支払います。要求は、選択した App Service プランに基づいて処理されます。App Service プランによって、ホストに利用されるハードウェアの量が決まります。たとえば、ハードウェアが専用に共有か、および予約されるメモリの量が、プランによって決定されます。"無料" のプランも用意されており、トラフィックの少ない小規模のサイトに使用できます。

アプリ サービスの種類

App Service を利用すると、次のような最も一般的なアプリ サービスのスタイルをホストできます。

- Web アプリ
- API アプリ
- WebJobs
- モバイル アプリ

App Service によって、Web からアクセス可能なアプリのホストでユーザーが扱うインフラストラクチャ関連の決定の多くが処理されます。

- デプロイと管理はプラットフォームに統合されます。
- エンドポイントはセキュリティで保護できます。
- サイトは、高トラフィック負荷を処理するため迅速にスケーリングすることができます。
- 組み込みの負荷分散機能とトラフィック管理機能によって可用性が高くなります。

このようなアプリのスタイルはすべて、同じインフラストラクチャでホストされ、以上のような利点が共有されます。この柔軟性により、App Service は、Web 指向アプリケーションをホストする理想的な選択肢となります。

Web アプリ

App Service には、ASP.NET、ASP.NET Core、Java、Ruby、Node.js、PHP、または Python を使用し、Web アプリをホストするための完全サポートが含まれます。ホスト オペレーション システムとして Windows か Linux のいずれかを選択できます。

API アプリ

Web サイトをホストする場合と同様に、選択した言語とフレームワークを利用し、REST ベースの Web API を構築できます。Swagger の完全サポートを受けられるほか、API をパッケージ化し、Azure Marketplace で公開できます。作成されたアプリは、HTTP または HTTPS ベースの任意のクライアントから利用できます。

WebJobs

WebJobs を使用すると、プログラム (.exe、Java、PHP、Python、または Node.js) やスクリプト (.cmd、.bat、PowerShell、または Bash) を Web アプリ、API アプリ、モバイル アプリと同じコンテキストで実行できます。それらは、スケジュールしたりトリガーで実行したりすることができます。多くの場合、WebJobs は、アプリケーション ロジックの一環としてバックグラウンド タスクを実行するために使用されます。

モバイル アプリ

App Service の Mobile Apps 機能を使用し、iOS アプリや Android アプリのバックエンドをすばやく構築します。Azure portal で数回クリックするだけで次を実行できます。

- クラウドベースの SQL データベースにモバイル アプリ データを格納する。
- MSA、Google、Twitter、Facebook など、一般的なソーシャル プロバイダーに対して顧客を認証する。
- プッシュ通知を送信する。
- C# または Node.js でカスタム バックエンド ロジックを実行する。

モバイル アプリ側には、ネイティブ iOS アプリ、ネイティブ Android アプリ、Xamarin アプリ、React ネイティブ アプリに対する SDK のサポートがあります。