**Web Apps overview**

Azure App Service Web Apps (or just Web Apps) is a service for hosting web applications, REST APIs, and mobile back ends. You can develop in your favorite language, be it .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python. You can run and scale apps with ease on Windows or Linux VMs (see [App Service on Linux](https://docs.microsoft.com/en-in/azure/app-service/containers/app-service-linux-intro)).

Web Apps not only adds the power of Microsoft Azure to your application, such as security, load balancing, autoscaling, and automated management. You can also take advantage of its DevOps capabilities, such as continuous deployment from VSTS, GitHub, Docker Hub, and other sources, package management, staging environments, custom domain, and SSL certificates.

With App Service, you pay for the Azure compute resources you use. The compute resources you use is determined by the App Service plan that you run your Web Apps on. For more information, see [App Service plans in Azure Web Apps](https://docs.microsoft.com/en-in/azure/app-service/azure-web-sites-web-hosting-plans-in-depth-overview).

The following 5-minute video introduces Azure App Service Web Apps.

**Why use Web Apps?**

Here are some key features of App Service Web Apps:

* **Multiple languages and frameworks** - Web Apps has first-class support for ASP.NET, ASP.NET Core, Java, Ruby, Node.js, PHP, or Python. You can also run [PowerShell and other scripts or executables](https://docs.microsoft.com/en-in/azure/app-service/web-sites-create-web-jobs) as background services.
* **DevOps optimization** - Set up [continuous integration and deployment](https://docs.microsoft.com/en-in/azure/app-service/app-service-continuous-deployment) with Visual Studio Team Services, GitHub, BitBucket, Docker Hub, or Azure Container Service. Promote updates through [test and staging environments](https://docs.microsoft.com/en-in/azure/app-service/web-sites-staged-publishing). Manage your apps in Web Apps by using [Azure PowerShell](https://docs.microsoft.com/en-us/powershell/azureps-cmdlets-docs) or the [cross-platform command-line interface (CLI)](https://docs.microsoft.com/en-us/cli/azure/install-azure-cli).
* **Global scale with high availability** - Scale [up](https://docs.microsoft.com/en-in/azure/app-service/web-sites-scale) or [out](https://docs.microsoft.com/en-in/azure/monitoring-and-diagnostics/insights-how-to-scale) manually or automatically. Host your apps anywhere in Microsoft's global datacenter infrastructure, and the App Service [SLA](https://azure.microsoft.com/support/legal/sla/app-service/)promises high availability.
* **Connections to SaaS platforms and on-premises data** - Choose from more than 50 [connectors](https://docs.microsoft.com/en-in/azure/connectors/apis-list) for enterprise systems (such as SAP), SaaS services (such as Salesforce), and internet services (such as Facebook). Access on-premises data using [Hybrid Connections](https://docs.microsoft.com/en-in/azure/biztalk-services/integration-hybrid-connection-overview) and [Azure Virtual Networks](https://docs.microsoft.com/en-in/azure/app-service/web-sites-integrate-with-vnet).
* **Security and compliance** - App Service is [ISO, SOC, and PCI compliant](https://www.microsoft.com/TrustCenter/). Authenticate users with [Azure Active Directory](https://docs.microsoft.com/en-in/azure/app-service/app-service-mobile-how-to-configure-active-directory-authentication) or with social login ([Google](https://docs.microsoft.com/en-in/azure/app-service/app-service-mobile-how-to-configure-google-authentication), [Facebook](https://docs.microsoft.com/en-in/azure/app-service/app-service-mobile-how-to-configure-facebook-authentication), [Twitter](https://docs.microsoft.com/en-in/azure/app-service/app-service-mobile-how-to-configure-twitter-authentication), and [Microsoft](https://docs.microsoft.com/en-in/azure/app-service/app-service-mobile-how-to-configure-microsoft-authentication)). Create [IP address restrictions](https://docs.microsoft.com/en-in/azure/app-service/app-service-ip-restrictions) and [manage service identities](https://docs.microsoft.com/en-in/azure/app-service/app-service-managed-service-identity).
* **Application templates** - Choose from an extensive list of application templates in the [Azure Marketplace](https://azure.microsoft.com/marketplace/), such as WordPress, Joomla, and Drupal.
* **Visual Studio integration** - Dedicated tools in Visual Studio streamline the work of creating, deploying, and debugging.
* **API and mobile features** - Web Apps provides turn-key CORS support for RESTful API scenarios, and simplifies mobile app scenarios by enabling authentication, offline data sync, push notifications, and more.
* **Serverless code** - Run a code snippet or script on-demand without having to explicitly provision or manage infrastructure, and pay only for the compute time your code actually uses (see [Azure Functions](https://docs.microsoft.com/en-us/azure/azure-functions/)).

Besides Web Apps in App Service, Azure offers other services that can be used for hosting websites and web applications. For most scenarios, Web Apps is the best choice. For microservice architecture, consider [Service Fabric](https://azure.microsoft.com/documentation/services/service-fabric). If you need more control over the VMs that your code runs on, consider [Azure Virtual Machines](https://azure.microsoft.com/documentation/services/virtual-machines/). For more information about how to choose between these Azure services, see [Azure App Service, Virtual Machines, Service Fabric, and Cloud Services comparison](https://docs.microsoft.com/en-in/azure/app-service/choose-web-site-cloud-service-vm).