| **Term** | **Definition** |
| --- | --- |
| **API** | Application Programming Interface |
| **Application modernization** | Helps organizations accelerate their digital transformation, take advantage of new technologies and services, and become more responsive to changing market dynamics |
| **Cloud native application** | An application developed from the outset to work only in the cloud environment, or an existing app that has been refactored and reconfigured with cloud native principles |
| **Continuous delivery** | Delivering small, well-designed, high-quality increments of software to customers |
| **Continuous deployment** | Progressing each new packaged build through the deployment lifecycle as rapidly as possible |
| **Continuous integration** | Creating packaged builds of the code changes released as immutable images |
| **Continuous monitoring** | Monitoring with tools that help developers understand the performance and availability of their applications, even before they’re deployed to production |
| **Delivery pipeline** | An automated sequence of steps that involves the stages of Ideation, Coding, Building, Deploying, Managing, and Continuous Improvement |
| **DevOps** | Collaborative approach where business owners and the development, operations, and quality assurance teams collaborate to continuously deliver software |
| **Hybrid multicloud** | An open standards-based stack that can be deployed on any public cloud infrastructure |
| **Microservices** | Break down large applications into their core functions |
| **Microservices architecture** | Approach in which a single application is composed of many loosely coupled and independently deployable, smaller components or services |
| **Monolithic architecture** | Approach in which a single application is built out of one piece of software |
| **Serverless** | Approach to computing that offloads responsibility for common infrastructure management tasks |
| **Service discovery** | Creates a roadmap for microservices to communicate |