Projektbeschreibung Zugverwaltung

Gruppe: J.Hofer, M.Rudlof, E.Stoisser

Die Anforderungen an das Cloud-Computing-Abschlussprojekt wurden am Beispiel einer Zugverwaltung umgesetzt.

# Azure-Automatisierung

Um die Umgebung, für die das Projekt automatisiert erstellen zu können, wurde das Powershell-Script „CreateEverything.ps1“ geschaffen. Es erfolgt die Erstellung von folgenden Azure-Komponenten:

* VM
* Ressource-Group
* Webservice & Plan
* Storage
* Cosmos-DB mit zwei Containern
* Functions

## Technologiebeschreibung

Im Nachfolgenden wird auf die technologischen Überlegungen lt. Aufgabenstellung eingegangen. Als nicht in der Vorlesung angesprochene Technologie wurde die exemplarische Anlage eine VM im PS-Script verwendet.

### Virtuelle Maschinen und virtuelle Netzwerke

Anders als in diesem Beispiel, hätte auch eine Umgebung von virtuellen Maschinen und einem „dazwischenliegenden“ virtuellen Netzwerk aufgebaut werden können. Wie im nächsten Abschnitt ersichtlich ist, hätte sich damit die die Ausprägungsstufe auf IaaS-Niveau verändert, da nun auch die virtuellen Computer und die darauf betriebene Software (zB SQL-Server) durch den Kunden verwaltet werden müsste.

### Cloud-Computing Ausprägungsstufen

Bei Cloudcomputing kann im Allgemeinen von einer Abstufung in der Ausprägung der Nutzung von Cloud-Diensten ausgegangen werden. Diese erstrecken sich von On-site, also der ausschließlich lokalen Nutzung von IT-Ressourcen, bis hin zur vollständigen Auslagerung aller Dienste in die Cloud.

Wie in der nachfolgenden Abbildung ersichtlich, werden die drei Ausprägungen mit Cloudanteil als IaaS, Paas und SaaS bezeichnet:

Graphical user interface

Description automatically generated

Abbildung 1: XaaS-Hautpmodelle

#### IaaS – Infrastructure as a Service

Hier wird bis ist der Übergabepunkt ein virtueller Computer. Der Kunde ist selbst für die Verwaltung aller darüberliegenden Komponenten verantwortlich. Auch Betriebssystem-Updates müssen durch den Kunden vorgenommen werden. Die darunterliegende „physische“ Welt ist aus Kundensicht jedoch nicht relevant. Es werden lediglich Leistungsausprägungen und Verfügbarkeiten gebucht.

#### PaaS – Platform as a Service

Der Kunde bucht die gesamte Plattform, auf der er seine Applikation ausrollt. Die Verwaltung von allen darunterliegenden Komponenten, werden durch den Dienstleister übernommen.

#### SaaS – Software as a Service

Der Kunde bezieht direkt und ausschließlich die von ihm gewünschte Software als Dienstleistung. Es muss keinerlei Verwaltungstätigkeit für Infrastruktur durch den Kunden vorgenommen werden.

Die Umsetzung dieses Projektes entspricht der Ausprägung PaaS.

# Umsetzung

In diesem Kapitel wird auf die eingesetzten Technologien näher eingegangen.

## Datenmodelle

Nachfolgend ist das Datenmodell als Screenshot der Train-Klasse dargestellt:

## Text Description automatically generated

Abbildung 2: Train-Klasse

## Storage

* TrainBlob
* TrainQueue

Als Monitoring-Lösung wird Application Insights verwendet.

## Cosmos DB

#### Partition-Key

public string RowKey { get; set; }

public string PartitionKey { get; set; }

#### Consitency-Level

Azure Cosmos DB bietet 5 unterschiedliche Konsistenz-Niveaus, oder in andren Worten, welche Datenstand mindestens zu einem bestimmten Zeitpunkt in unterschiedlichen Azure-Regionen verfügbar ist.

* Strong

User haben die Garantie aus allen Regionen den neusten Schreibvorgang zu lesen, für den ein Commit ausgeführt wurde.

* Bounded staleness

User aus anderen Regionen lesen Schreibvorgänge verzögert (max. 5 s, oder 100 Versionen), bekommen diese aber in derselben Reihenfolge gelesen.

* Session

Innerhalb einer Session mit „Strong“ vergleichbar, Regionen außerhalb der Session bekommen Schreibvorgänge verzögert, jedoch in derselben Reihenfolge angezeigt.

* Consistent prefix

Clients anderer Regionen erhalten ein bestimmtes Prefix von Update ohne Lücken, die Reihenfolge wird dabei immer eingehalten.

* Eventual

Updates werden in anderen Regionen verzögert und in beliebiger Reihenfolge angezeigt.

Im PS-Script wird bei der Erstellung des Service die Einstellung “$consistencyLevel = "Session"“ verwendet. Dies ermöglicht innerhalb unserer Session volle Konsistenz und wird für im Benutzerkontext geschriebene Anwendungen empfohlen.

## Azure Functions

Im Projekt wurden zwei Azure-Functions eingesetzt:

1. ichmagzuegefunction1

Dient beim Deployment der queue-Funktionalität als Service.

1. ichmagzuegefunction2

Dient beim Deployment der Blob-Funktionalität als Service.

### Serverless computing

Hierbei handelt es sich um eine voll virtualisierte Umgebung. Die Infrastruktur von die Code-Ausführung wird aus Kundensicht „unsichtbar“ zur Verfügung gestellt. Ressourcen können dynamisch skaliert werden und so wechselten Anforderungen gerecht werden. Da im Hintergrund jedoch naturgemäß klassische Server zum Einsatz kommen, kann dies zu Sicherheits- (Meltdown), oder Verfügbarkeitsproblemen (DDoS-Attacken auf ein anderes Service am selben Server) führen.

# Ausarbeitung eLearning

## Azure Key Vault

Dient zur sicheren Speicherung von beispielswiese API-Keys, oder allgemein, Geheimnissen. So kann einfach global auf Schlüssel zugegriffen werden, ohne den Code dafür selbst schreiben zu müssen. Dadurch sinkt das Risiko von Fehlern in der Implementierung dieser Funktionalität. Weiters baut darauf „Azure Managed Identities“ auf. Ein Service, das genutzt werden kann, um Azure AD-Tokens zu beziehen und so die Komplexität der Credential-Verwaltung innerhalb von Applikation minimiert.

Virtuelle Maschinen

#### Welche Punkte müssen bei der Wahl des Standorts müssen berücksichtigt werden?

* Erfüllung aller Complience-Anforderungen (z.B. DSGVO)
* Verfügbarkeit der Benötigten Dienste
* Kosten

#### Was ist eine Azure-Ressource?

Ein einzelnes Azure-Service wie z.B. Storage oder AD.

*Welchem Zweck dienen Virtuelle Netzwerke (VNETs) im Zusammenhang mit virtuellen Azure-Computern?*

Ermöglichen die virtuelle, private Vernetzung zwischen Azure-VMs, aber auch mit on-premise-Netzwerken. Somit können beispielsweise Azure-VMs sicher an eine bestehen Netzwerkinfrastruktur angebunden werden.

*Welche zwei unterschiedlichen Kosten im Zusammenhang mit VMs können anfallen?*

Verfügbarkeits- und leistungsbezogene Kosten.

*Welche Optionen zum Erstellen und Verwalten von VMs können verwendet werden?*

* Windows-Admin-Center
* PS
* CLI

*Was ist eine Verfügbarkeitsgruppe?*

Verfügbarkeitsgruppen dienen zur Konfiguration von Redundanzen und Verfügbarkeiten.

*Was ist RDP? In welchem Zusammenhang stehen RDP und Azure VMs*

RPD ist das Remote-Desktop-Protocol und er ermöglich den Zugriff auf einen Windows-Desktops aus der Ferne. In Zusammenhang mit Azure-VMs ist es die einzige Möglichkeit, um direkt auf den Server zuzugreifen, da ein physischer Zugang nicht möglich ist.

## Application Insights

Ist ein Feature von Azure Monitor und ermöglich die Überwachung von Live-Web-Apps.

* Application Map

Ermöglicht die Erkennung von Performance-Problemen über alle Komponenten einer verteilten Anwendung.

* Smart Detection

Erkennt und warnt automatisch, wenn Performance-Probleme und außergewöhnliches Verhalten einer Applikation erkannt werden.

* Live Metrics Stream

Ist ein Monitoring-Cockpit zur Überwachung von KPIs

* Alerts

Es können „Alerts“ konfiguriert werden, die bei bestimmten Problemen versandt werden. (Siehe auch Smart Detection)

* Availability

Dient zur Konfiguration unterschiedlicher Verfügbarkeitsprüfungen. Bei Unterschreitung kann ein Alter konfiguriert werden.

* Performance und Analytics

Telemetriedaten können, durch Integration des Service, direkt in der Applikation erhoben werden.