



Abschlussprüfung Sommer 2026

Fachinformatiker Fachrichtung: Systemintegration

Dokumentation zur betrieblichen Projektarbeit

Integration einer Azure-KI zur Dokumentenverarbeitung

**Planung und Integration einer Azure-basierten KI-Lösung zur automatisierten
Bearbeitung von Kundendokumenten.**

Abgabedatum: 09.06.2026

Prüfungsbewerber:

Ben Borchert

Identnummer:

Ausbildungsbetrieb:

Marabu GmbH & Co. KG

Asperger Straße 4

71732 Tamm



Inhaltsverzeichnis	
Inhaltsverzeichnis	2
Abbildungsverzeichnis	3
Tabellenverzeichnis	4
Glossar	5
1 Einleitung	6
1.1 Ausgangssituation	6
1.2 Projektabgrenzung	6
1.3 Projektumfeld	6
1.4 Projektbegründung	7
1.5 Schnittstellen	7
2 Projektplanung	8
2.1 Ablaufplanung	8
2.2 Ressourcenplanung	9
2.4 Wirtschaftlichkeitsrechnung	10
2.4.1 Return on Investment	10
2.4.2 Amortisationsrechnung	11
3 Projektdurchführung	12
3.1 Analyse des Ist-Zustands	12
3.2 Analyse des Soll-Konzepts	12
3.3 Umsetzung	12
3.3.1 DevOps einrichten	12
3.3.2 Azure Ressourcen anlegen	13
3.3.3 Azure Web-App an Unternehmensnetzwerk anbinden	14
3.3.4 Zugriffsregeln über die Firewall	16
3.4 Testphase & Qualitätssicherung	19
4 Projektabschluss	21
4.1 Projektrückblick	21
4.2 Projektaussicht	21
4.3 Dokumentationen	21
5 Anhang	22
5.1 Zeitplanung	22
5.2 Gantt-Diagramm Arbeitsphasen	24
5.3 DevOps & Azure Screenshots	24
5.4 Quellenangabe	32

Abbildungsverzeichnis

Abbildungsverzeichnis

Abbildung 1: DNS Eintrag erstellen	15
Abbildung 2: Zertifikats-Skript Nr.1	15
Abbildung 3: Zertifikats-Skript Nr. 2	16
Abbildung 4: Firewall Objekt anlegen	16
Abbildung 5: Firewall Regel - BranchesToWebApp.....	17
Abbildung 6: Firewall Regel - DEToWebApp.....	17
Abbildung 7: Firewall Regel - BranchesClientsToAzure	18
Abbildung 8: Firewall Regel - AzureToClientsSE	18
Abbildung 9: ASP Preistarife	20
Abbildung 10: DevOps Userberechtigung	24
Abbildung 11: DevOps Work-Item	25
Abbildung 12: DevOps Kanban-Board	25
Abbildung 13: Azure Ressourcengruppe anlegen	26
Abbildung 14: ACR Ressource	26
Abbildung 15: ACR erstellen.....	26
Abbildung 16: ASP Ressource	27
Abbildung 17: ASP anlegen	27
Abbildung 18: Web Service Ressource	28
Abbildung 19: Web-App Basic settings	28
Abbildung 20: Web-App Container settings	29
Abbildung 21: Foundry Ressource	29
Abbildung 22: Foundry anlegen	29
Abbildung 23: LLM model deployment.....	30
Abbildung 24: LLM deployment Konfiguration	30
Abbildung 25: Private EndPoint anlegen.....	31
Abbildung 26: Custom Domain einrichten	31
Abbildung 27: .pfx-Zertifikat hochladen	32

Tabellenverzeichnis

Tabellenverzeichnis

Tabelle 1: Grobe Zeitplanung	8
Tabelle 2: Personalrechnung	10
Tabelle 3: Kostenrechnung im Projektumfeld	10
Tabelle 4: Testfall-Matrix	19
Tabelle 5: Detaillierte Zeitplanung	22
Tabelle 6: Realer Zeitplan	23
Tabelle 7: Gantt-Diagramm Arbeitsphasen.....	24

Glossar

Für die Projektdokumentation werden Firmen- und Mitarbeiternamen anonymisiert:

OneTech Dienstleister, der Web-App programmiert hat.

MAPA Kunde, dessen Dokumente für die Entwicklung der Web-App verwendet wurden.

Fr. Sol Mitarbeiterin der Fachabteilung in Schweden.

Sonstige Abkürzungen:

KI Künstliche Intelligenz

NDA Non-Disclosure Agreement (Vertraulichkeitsvereinbarung)

DSGVO Datenschutzgrundverordnung

BEP Break-Even-Point

ACR Azure Container Registry

ASP Azure Service Plan

LLM Large Language Model

DMZ Demilitarisierte Zone

LAN Local Area Network

VNet Virtual Network

NIC Network Interface Card

DNS Domain Name System

TXT-Record Text-Eintrag in DNS

.pfx-Datei Personal Information Exchange file

HTTPS Hypertext Transfer Protocol

VLAN Virtual Local Area Network

VM Virtual Machine

Nr Nummer

Begriffsdefinitionen:

Subscription Logischer Container, um die Abrechnung in Azure zu verwalten

Repository Zentraler Speicherort für projektrelevante Dokumente

Image Vorlage, zur Erstellung einer VM/ eines Containers

Priv. EndPoint .. Private IP-Adresse in VNet, über die der Dienst sicher erreichbar ist

Peering Direkte, private Verbindung zwischen zwei VNets

Analyzer Tool zur Sammlung, Auswertung und Analyse von (Firewall-) Logs und Ereignissen

EntraID Cloudbasierter Identitäts- und Zugriffsmanagement-Dienst von Microsoft

1 Einleitung

1.1 Ausgangssituation

Das Projekt wird im betrieblichen Umfeld der Marabu GmbH & Co. KG absolviert, welches ein internationales Chemieunternehmen ist und Farben in zwei Kategorien produziert: Druckfarben für die Industrie und Kreativfarben für den Privatgebrauch. Das Unternehmen umfasst weltweit ca. 500 Angestellte, darunter arbeiten knapp 250 Mitarbeiter in Deutschland (Hauptsitz in Tamm). Die im Abschlussprojekt enthaltenen Aufgaben beziehen sich auf den Projektwunsch einer bestimmten Fachabteilung in Schweden. Diese muss Kundendokumente manuell über den Adobe Illustrator bearbeiten und korrigieren, bevor sie diese weiterverwenden können. Die Projektanfrage beinhaltet nun die Optimierung dieses Prozesses mit Hilfe eines KI-Agenten, der die Nachbearbeitung automatisieren soll. Der KI-Agent wird durch den Dienstleister OneTech als Web-App zur Verfügung gestellt und von mir in die Microsoft Azure Infrastruktur des Unternehmens aufgenommen. Im Rahmen meines Abschlussprojektes übernehme ich die Projektleitung und die technische Umsetzung auf Azure-Ebene.

1.2 Projektabgrenzung

Die erhaltenen Dokumente unterscheiden sich je nach Kunde. Um ein klares und messbares Ziel zu erhalten liegt der Projektfokus auf einem spezifischen Kunden (MAPA), diese Klarstellung gilt vor allem für OneTech, da sie den KI-Agenten zur Verfügung stellen. Meine Aufgabenbereiche beinhalten keinerlei Kontaktpunkte mit der Programmierung der KI, dementsprechend wird diese im weiteren Vorgehen als Black-Box betrachtet und nicht weiter vertieft. Ich bin ausschließlich für die Bereitstellung und Konfiguration der Azure Subscription verantwortlich, sowie das Leiten des Projektes in Bezug auf Kommunikation, Kontrolle und Wirtschaftlichkeit.

1.3 Projektumfeld

Trotz der internationalen Präsenz wird die vollständige IT-Infrastruktur des Unternehmens in einer zentralen IT-Abteilung verwaltet. Hier wird eine hybride IT-Infrastruktur betrieben, das bedeutet, dass sowohl On-Premise gehostete, als auch Cloud-basierte Dienste in Betrieb sind. In den letzten Jahren gibt es jedoch einen technischen Aufschwung, dieser entwickelt das Unternehmen zunehmend zu einem Cloud-Getriebenen. Im Zuge der Digitalisierung gewinnt auch der Einsatz von KI mehr an Bedeutung, so entstand eben auch der Grundgedanke dieses Projektes. Das Unternehmen ist nun auch Teil einer Microsoft-Partnerschaft, das bedeutet für das Projekt, dass einige der Kosten von Microsoft übernommen werden (Dazu jedoch mehr in der Ressourcenplanung). Der Kunde ist in diesem Fall die interne Fachabteilung, somit bezieht sich die spätere Kundendokumentation ebenfalls auf die Mitarbeiter dieser Abteilung. Azure ist für das Unternehmen nichts Neues und in anderen Bereichen bereits in Verwendung, daher fällt mir die Systemwahl zur Anbindung der von OneTech entwickelten Web-App sehr leicht. Die vollständige Organisation der Arbeitsschritte und -verteilung läuft über DevOps, auch hier handelt es sich wieder um ein Tool, das bereits vor dem Projekt verwendet wird. Die Kommunikation über Teams, Email und DevOps ermöglicht es mir das Projekt vollständig aus dem Büro heraus zu bearbeiten.

Einleitung

1.4 Projektbegründung

Während der Ausbildung hatte ich gelegentlich Berührungspunkte mit Azure. Da der Dienst sehr umfangreich und vielfältig ist, habe ich stetig mehr Interesse an Azure gewonnen und möchte mich nun intensiver damit auseinandersetzen. Dazu kommt noch, dass ich bereits zuvor bei der Einführung eines anderen KI-Projektes beteiligt war und mich demnach mit den Anforderungen solcher Projekte gut auskenne. Mein persönliches Ziel des Projektes ist es, mir mehr Wissen im Azure-Dienst anzueignen, vor allem in den Bereichen der Sicherheit- und Berechtigungskonzepte.

1.5 Schnittstellen

Während des Projektes werde ich mich mit drei Parteien absprechen: Die betroffene Fachabteilung, der Dienstleister OneTech und meine internen IT-Kollegen (Teamleiter und System Administrator). Die Kommunikation im Projekt findet auf Englisch statt, da unser „Kunde“, die Fachabteilung, ihren Sitz in Schweden hat. Um die Kommunikation zu vereinfachen erhalten wir einen spezifischen Ansprechpartner aus Schweden (Fr. Sol). Sie wird uns regelmäßig Feedback zum System geben und organisiert die Kommunikation zu MAPA. Das ist in der Hinsicht wichtig, da es sich bei den Kundendokumenten um NDA-geschützte Inhalte handelt und man deshalb während der Konzipierung der Web-App besonderen Wert auf den Datenschutz legen muss. Neben den bereits genannten Diensten (Azure und DevOps) werden im Laufe des Projektes noch ein DNS, ein lokaler Server zur Erstellung von Zertifikaten und die Firewall verwendet.

2 Projektplanung

2.1 Ablaufplanung

Das Projekt wird, entgegen des im Antrag definierten Zeitraumes, nun am 15.05.26 starten und läuft über knapp zwei Wochen bis inklusive 29.05.26. In diesem Zeitraum fällt ein Feiertag an, der für die folgende Zeitplanung beachtet wird. Die Verschiebung des Projektes ist notwendig, da sich die Projekt-Annahme seitens der IHK am 07.05.26 ereignete (Projektstart laut Antrag jedoch 27.04.26) und wurde den Beteiligten bereits per Mail übermittelt. Das Projekt wird neben der alltäglichen Arbeit absolviert, das bedeutet, dass weiterhin die Arbeit im Ticketsystem und anderen Projekten stattfindet. Im Folgenden befindet sich nun der im Antrag erstellte Zeitplan, eine detaillierte Zeitplanung ist im Anhang zu sehen [Tabelle 5: Detaillierte Zeitplanung].

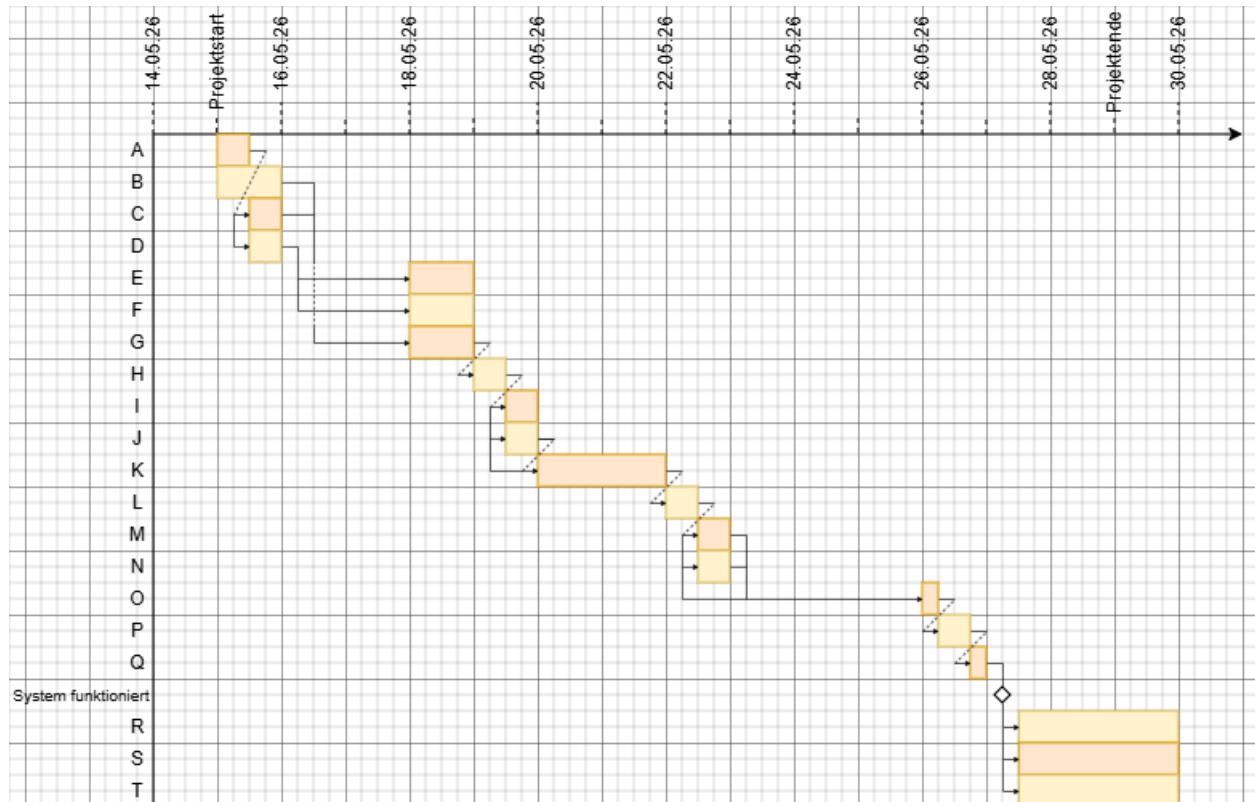
Projektphase	Geplante Zeit
Anforderungsanalyse	4h
Wirtschaftlichkeitsrechnung	2h
Projektplanung	4h
Technische Implementation	15h
Funktionstest mit Anwender	3h
Projektdokumentation	12h
Gesamt	40h

Tabelle 1: Grobe Zeitplanung

Während des Projektablaufes wird ein hybrides Projektmanagement verwendet. Einerseits liegt ein über DevOps organisiertes Kanban-Board vor, in dem meine Arbeitspakete definiert werden. Andererseits wurde sich dazu entschieden regelmäßige Meetings abzuhalten, um den aktuellen Stand des Projektes besprechen und User-Feedback einbeziehen zu können (Kanban + Agiles Projektmanagement).

Zur grafischen Abbildung der Zeitplanung des Projektes müssen zuerst Arbeitspakete definiert werden, damit der jeweilige Aufwand der Arbeitsphasen eingeschätzt werden kann. Wie bereits angesprochen erfolgt dies durch ein Kanban-Board, ein Bild dazu befindet sich im Anhang [Abbildung 12: DevOps Kanban-Board]. Die Entscheidung zur grafischen Abbildung fällt nun auf einen Gantt-Diagramm, da dieses eine leicht verständliche Übersicht aller Arbeitsphasen und deren Abhängigkeiten darstellt. Eine Übersicht der verwendeten Arbeitsphasen mit den jeweiligen Vorgängern und Nachfolgern ist im Anhang zu finden [Tabelle 7: Gantt-Diagramm Arbeitsphasen].

Projektplanung



2.2 Ressourcenplanung

Sowohl der Azure Dienst als auch DevOps basieren auf einer Microsoft Cloud-Umgebung, somit wird keine spezifische Hardware in Anspruch genommen. Dennoch benötigen wir für die Web-App eine eigene Azure-Subscription, da sie getrennt von unserer Standard-Subscription verwaltet wird. Die Trennung dieses Projektes erfolgt aufgrund der Abrechnung. Da für dieses Projekt nicht die IT-Abteilung, sondern die Kostenstelle der Fachabteilung belastet wird, sollte sie für eine einfache Abrechnung komplett getrennt sein. Der Preis dieser Subscription hängt mit dem Nutzungsgrad der Web-App zusammen. Desto mehr diese verwendet wird, desto mehr Kosten fallen an. Im Weiteren bewerte ich noch die Wirtschaftlichkeit des Projektes in Betracht der jährlichen Kosteneinsparung nach Projektabschluss. Benötigte Lizenzen für diese Systeme sind bereits vor dem Projekt in Einsatz, sie können in der Kostenrechnung vernachlässigt werden.

Neben meiner Wenigkeit gibt es noch zwei weitere IT-Kollegen, die mich bei Fragen unterstützen. Sowohl für diese beiden Kollegen, als auch für Fr. Sol werde ich mit einem Referenzwert des Gehaltes rechnen. Aufgrund der DSGVO ist es mir nicht erlaubt die Gehälter zu veröffentlichen, da es sich um personenbezogene Daten handelt. Die Programmierung seitens OneTech wird zu einem fixen Preis gekauft. Diese Kosten werden zwar nicht durch meine Aufgabenpakete verursacht, jedoch werde ich sie ebenfalls in der Kostenrechnung mit aufnehmen, da sie, genau wie mein Aufgabenbereich, zu einem gemeinsamen Großprojekt gehören. Dieser Fix-Preis wird durch die vorhandene Microsoft-Partnerschaft jedoch zum Teil von Microsoft selbst gezahlt.

Projektplanung

Beispiel zur Ausrechnung eines Stundenlohns (7,5 Arbeitsstunden pro Tag, 220 Werktage pro Jahr, 1.326,00 € Monatsgehalt):

$$7,5 \frac{h}{Tag} * 220 \frac{Tage}{Jahr} = 1.650 \frac{h}{Jahr}$$

$$1.326,00 \frac{€}{Monat} * 12 \frac{Monate}{Jahr} = 15.912 \frac{€}{Jahr}$$

$$\frac{15.912 \frac{€}{Jahr}}{1.650 \frac{h}{Jahr}} \approx 9,64 \frac{€}{h}$$

Mit dieser Formel werden nun die jeweiligen Kosten pro Person berechnet und in der Personalrechnung aufsummiert.

Person	Projektrolle	Arbeitsdauer [in h]	Referenzwert [pro Monat]	Kosten [pro h]	Kosten [im Projekt]
Auszubildender	Projektleiter	40	1.326,00 €	9,64 €	385,60 €
Teamleiter	Aushilfe	3	5.500,00 €	40,00 €	120,00 €
IT-Administrator	Aushilfe	4	3.775,00 €	27,45 €	109,80 €
OneTech	Stakeholder	0 (In Fix-Preis enthalten)	-	0,00 €	0,00 €
Fr. Sol	Stakeholder	5	4.500 €	32,73 €	163,65 €
Gesamt					779,05 €

Tabelle 2: Personalrechnung

Kostenart	Operator	Kosten
Personalkosten	+	779,05 €
Fix-Preis des Projektes	+	42.000 €
Erlass durch Microsoft	-	18.000 €
Azure App Service	+	14,05 €
Azure Container Registry	+	2,03 €
Azure Foundry Model	+	0,02 €
Gesamt		24.795,15 €

Tabelle 3: Kostenrechnung im Projektumfeld

2.4 Wirtschaftlichkeitsrechnung

2.4.1 Return on Investment

$$ROI = \frac{\text{Einsparung pro Jahr}}{\text{Investitionskosten}}$$

Der Return on Investment (ROI) zeigt als Kennzahl, wie viel Gewinn das Unternehmen im Verhältnis zum eingesetzten Kapital erzielt hat. Die Einsparungen lassen sich durch den vorherigen Aufwand der Fachabteilung ermitteln. In der Fachabteilung arbeiten insgesamt 3 Mitarbeiter, für die Rechnung wird jedoch der Aufwand von Fr. Sol [in h] als Referenzwert für beide ihrer Mitarbeiter verwendet (12h Aufwand pro Woche). Der Betrachtungszeitraum ist ein Jahr.

Projektplanung

$$12 \frac{h}{Woche} * 52 Wochen = 624 \frac{h}{Jahr}$$

$$624 h : 12 Monate = 52 \frac{h}{Monat}$$

Rechnungsweg für die jährlichen Einsparungen:

$$((\text{monatlicher Aufwand} * \text{Stundenlohn}) * \text{Mitarbeiteranzahl}) * 12$$

Mit den korrekten Werten ergibt sich folgende Rechnung:

$$\text{Einsparung pro Jahr} = \left(\left(52 \frac{h}{Monat} * 32,73 \text{ €} \right) * 3 \right) * 12 = 61.270,56 \text{ €}$$

Nach dem Projektabschluss ist ein Supportvertrag mit OneTech geplant, der dem Unternehmen 4h Support pro Monat zur Verfügung stellt (48h im Jahr), mit Kosten in Höhe von 150 € pro Stunde. Die Kosten durch diesen Supportvertrag werden im Jahreszeitraum nun auch beachtet:

$$\text{Supportkosten} = 48 h * 150 \frac{\text{€}}{h} = 7.200 \text{ €}$$

$$\text{Gesamtkosten} = 24.795,15 \text{ €} + 7.200 \text{ €} = 31.995,15 \text{ €}$$

$$ROI = \frac{61.270,56 \text{ €}}{31.995,15 \text{ €}} \approx 1,91$$

Im Betrachtungszeitraum von einem Jahr werde 191% Gewinn im Verhältnis zu den Kosten erzielt.

2.4.2 Amortisationsrechnung

Die Amortisationsrechnung berechnet die Dauer, bis sich das Projekt zurückgezahlt (amortisiert) hat.

$$BEP = \frac{\text{Investitionskosten}}{\text{Einsparung pro Jahr}}$$

$$BEP = \frac{31.995,15 \text{ €}}{61.270,56 \text{ €}} \approx 0,52$$

Wenn das System länger als 0,52 Jahre genutzt wird, ist es wirtschaftlich.

3 Projektdurchführung

3.1 Analyse des Ist-Zustands

Das Unternehmen verwendet Azure bereits in anderen Projekten, es existieren zurzeit vereinzelte Subscriptions von denen jedoch nur Eine produktiv verwendet wird. Für das Projekt wird eine neue Subscription von Grund auf angelegt. Dort erstelle ich alle nötigen Ressourcen und verbinde so die Web-App mit Azure. Die verwendete Firewall ist bereits in Betrieb, somit müssen nur noch die spezifischen Regeln und Routen angelegt werden, um die Web-App an das interne Netzwerk anzubinden. OneTech hat die Web-App bereits vollständig entwickelt und stellt uns Diese mit allen weiteren Informationen über eine DevOps-Repository zur Verfügung. Die Fachabteilung ist über den Projektablauf informiert und die gemeinsamen Meetings werden im Vorhinein terminiert.

3.2 Analyse des Soll-Konzepts

Die vorgelegte Web-App soll nach dem Projektende vollständig in Azure integriert sein. Durch die Anbindung an unser internes Netzwerk soll der externe Zugriff deaktiviert werden, während der interne Zugriff Standort-basierend eingeschränkt wird, das bedeutet, dass zum Projektende nur Schweden und Deutschland auf die Web-App zugreifen dürfen. Als Entwickler benötigt OneTech jedoch Zugriff auf das System, sie sollen planmäßig Zugriff auf das im Unternehmen vorhandene VPN-Portal erhalten. So können sie die Web-App weiterhin entwickeln und testen. Es soll außerdem eine spezifische Domain hinterlegt werden, über welche die Web-App aufrufbar ist. Die monatlichen Kosten sollen minimiert werden, das bedeutet in Bezug auf die VM, dass sie nicht über-skaliert werden darf. Das Produkt soll nach Projektende einen Stand erreichen, der anschließende Folge-Projekte ermöglichen soll. Demnach muss die Infrastruktur so geplant werden, dass sie weltweit für jede Tochtergesellschaft anwendbar und in Bezug auf Leistung und Funktionen der Web-App sowohl vertikal als auch horizontal skalierbar ist. Außerdem muss das Projekt sauber dokumentiert werden. Dazu gehört nicht nur die Dokumentation der Durchführung und Planung, sondern auch eine User-Anleitung für die Fachabteilung und ein Maintenance-Guide für meine IT-Kollegen. So soll sichergestellt werden, dass das Projekt auch nach Abschluss meiner Projektarbeit reibungslos läuft und Fehlverhalten behoben werden.

3.3 Umsetzung

3.3.1 DevOps einrichten

Das Unternehmen verwaltet bereits eine Organisation in DevOps, das neue Projekt kann dementsprechend in der bestehenden Organisation erstellt werden. Die User werden nun auf Organisationsebene hinzugefügt und für das betroffene Projekt berechtigt. Die dort verwalteten Aufgaben betreffen nur mich und OneTech, somit müssen nur die externen Accounts von OneTech als „Project Contributors“ hinzugefügt werden. Diese Benutzerrolle ist zwar die niedrigste aller Rollen, deckt jedoch die benötigten Funktionen und Rechte vollständig ab.

Als nächstes können nun alle Aufgabenpakete definiert und den verantwortlichen Personen zugewiesen werden. Daraus entsteht ein in vier Kategorien aufgeteiltes Kanban-Board: *New*, *In Progress*, *Committed*, *Done*.

[Abbildung 10: DevOps Userberechtigung]

[Abbildung 11: DevOps Work-Item]

[Abbildung 12: DevOps Kanban-Board]

Projektdurchführung

3.3.2 Azure Ressourcen anlegen

Um eine neue Azure-Subscription anzulegen benötige ich meinen Teamleiter, dieser besitzt die notwendigen Berechtigungen. Die neue Plain-Subscription wird im weiteren Vorgehen nun von mir eingerichtet.

Zunächst wird eine Ressourcengruppe angelegt. Die Ressourcengruppe ist wie ein Container, in dem alle Ressourcen erstellt und verwaltet werden. Beachten muss man hierbei nur, dass die korrekte Region ausgewählt wird. Manche Ressourcen stehen z.B. nur in bestimmten Regionen zur Verfügung. Für das Projekt wird für alle Ressourcen die Region „Sweden Central“ ausgewählt. Einerseits ist die Latenz der schwedischen Server bei unseren schwedischen Mitarbeitern am geringsten, andererseits werden vertrauliche Daten verarbeitet, wofür sich der EU-Raum durch die gesetzlichen Regelungen am besten eignet.

[Abbildung 13: Azure Ressourcengruppe anlegen]

Nun werden die Ressourcen nacheinander angelegt, angefangen mit der sogenannten Azure Container Registry (ACR). Die VM, über welche später die Web-App gehostet wird, benötigt zur Erstellung ein Image. Dieses Image hat OneTech aus einem Docker-File gebaut und uns zur Verfügung gestellt. Der Zweck des ACR ist es nun, dieses Image an Azure anzubinden und anderen Ressourcen zur Verfügung zu stellen.

[Abbildung 14: ACR Ressource]

[Abbildung 15: ACR erstellen]

Sobald das Image in Azure angebinden ist, kann die VM erstellt werden. Dazu muss ein App Service Plan (ASP) angelegt werden. Verwendet wird eine Linux-VM mit einem „Basic B1“-Tarifplan. Sollte man später beim Testen feststellen, dass dieser Tarifplan nicht ausreicht, kann er im Nachgang noch gewechselt werden.

[Abbildung 16: ASP Ressource]

[Abbildung 17: ASP anlegen]

Als Drittes wird nun die eigentliche Web-App als Azure Web Service angelegt.

[Abbildung 18: Web Service Ressource]

[Abbildung 19: Web-App Basic settings]

[Abbildung 20: Web-App Container settings]

Das LLM wird innerhalb eines Microsoft Foundry angelegt. Ein Foundry ist eine zentrale Plattform zur Verwaltung von KI-Anwendungen und -Modellen und vergleichbar mit einem Container für mehrere einzelne Foundry Projects. Das Foundry Project ist ein isolierter Arbeitsbereich, also ein konkreter Workspace für die Konfiguration des LLMs.

[Abbildung 21: Foundry Ressource]

[Abbildung 22: Foundry anlegen]

Projektdurchführung

Über das Foundry Portal kann nun ein gewünschtes KI-Modell ausgewählt werden. Für das Projekt entscheidet man sich vorerst für die Version „gpt-5.2-chat“. Wie auch bei der VM zuvor, kann nach Auswertung der User-Tests ein anderes passenderes Modell verwendet werden. Die tatsächliche Effektivität des LLMs lässt sich im Vergleich zur VM nur schwer identifizieren, da das LLM bei der Verarbeitung eines Dokumentes in der Web-App wenige Aufgaben durchführt und kaum ausgelastet wird. Durch diese geringe Verwendung fallen die Kosten für das LLM entsprechend klein aus. (ca. 0,02€ nach 8 Tagen)

[Abbildung 23: LLM model deployment]

[Abbildung 24: LLM deployment Konfiguration]

3.3.3 Azure Web-App an Unternehmensnetzwerk anbinden

Bisher ist die Web-App über einen vorgefertigten Azure-Link öffentlich erreichbar, das wird nun unterbunden indem die Netzwerk-Einstellung der Ressource auf „Disable public access“ gesetzt wird. Bevor die Web-App an das Unternehmensnetzwerk angebinden wird, muss entschieden werden, ob sie in die DMZ oder ins LAN aufgenommen wird. Die Aufnahme in die DMZ würde bedeuten, dass die Web-App weiterhin aus dem Internet erreichbar ist. Da jedoch nur die betroffene Fachabteilung produktiv mit dem System arbeitet, entscheide ich mich gegen die Anbindung innerhalb der DMZ. Stattdessen wird die Web-App ins LAN aufgenommen, sodass sie nur innerhalb des Unternehmensnetzwerkes aufrufbar ist. Das LAN besteht in Azure bereits als ein eigenständiges VNet. Die Verbindung von der Web-App ins LAN-VNet wird mit Hilfe eines Private EndPoints hergestellt. Dieser ermöglicht es, dass Cloud-Anwendungen wie z.B. die Web-App über eine virtuelle NIC auf andere VNets zugreifen können.

[Abbildung 25: Private EndPoint anlegen]

Die Web-App ist die Einzige der Ressourcen, die mit anderen Netzwerken kommunizieren muss. Ein Private EndPoint genügt somit, um eine reibungslose Anbindung zum Unternehmensnetzwerk herzustellen. Bei zwei oder mehr kommunikationsfähigen Ressourcen würde man ein eigenes VNet mit Subnetzen für die Ressourcengruppe anlegen und die beiden VNets über ein Peering der Subnetze verknüpfen. Diese Variante ist für das Projektumfeld nicht relevant, kann jedoch in Betracht eines möglichen Nachfolgeprojektes beachtet werden.

Als Nächstes geht es um das Hinterlegen einer Custom Domain. In Azure muss die angegebene Domain erst validiert werden, bevor sie hinzugefügt und mit einem Zertifikat hinterlegt werden kann. Das bedeutet, dass Azure zuerst sicherstellt, dass die Domain zum Unternehmen gehört, bevor sie hinterlegt wird. Um die Domain validieren zu können, muss sie im DNS des Unternehmens eingetragen werden. Dort wird nun ein neuer Eintrag vom Typ TXT-Record angelegt. Sowohl der Name als auch der eingetragene Wert werden von Azure vorgeschrieben und müssen nur kopiert werden.

Projektdurchführung

Zone bearbeiten - marabu.com

Hier kannst du deine Zone bearbeiten. Du kannst z. B. Resource-Records und Subdomains anlegen und verändern und die Nameserver-Einstellung der Zone konfigurieren.

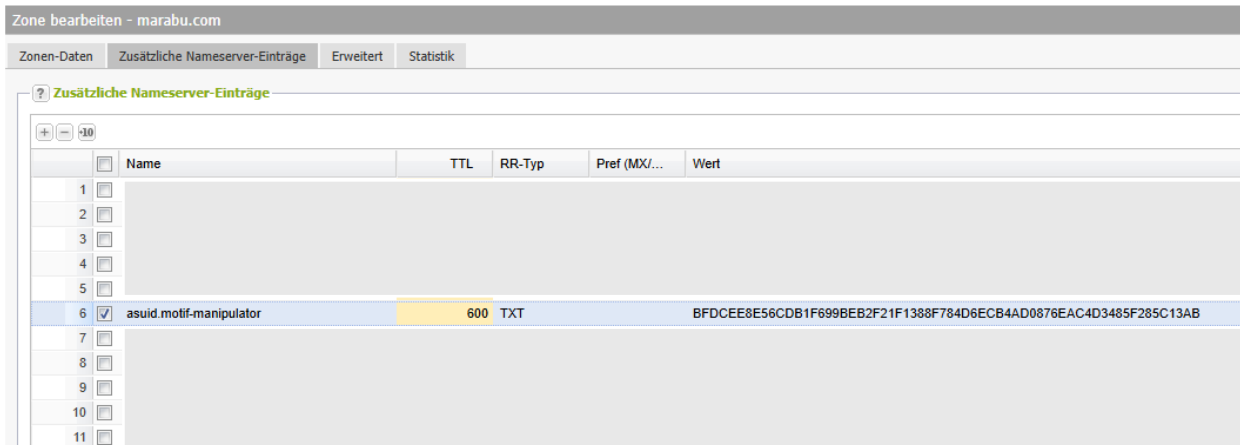


Abbildung 1: DNS Eintrag erstellen

[Abbildung 26: Custom Domain einrichten]

Die Web-App ist nun im LAN über die hinterlegte Domain aufrufbar, jedoch nur über das Protokoll HTTP. Um die Verbindung nun sicher zu gestalten wird ein gültiges Zertifikat benötigt. Dieses wird über einen Zertifikats-Server im Unternehmen erstellt. Dort ist ein Template hinterlegt, das als Kopie bearbeitet wird. Die kopierten Inhalte des Templates werden auf die neue Domain angepasst, so müssen beispielsweise Variablen-Werte innerhalb zweier Batch-Skripte geändert werden.

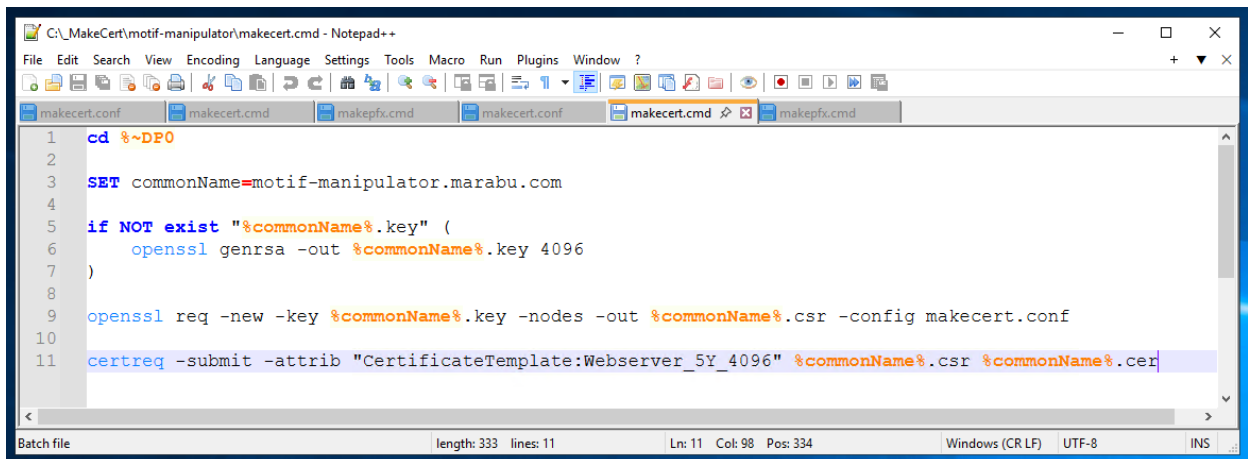


Abbildung 2: Zertifikats-Skript Nr.1

Projektdurchführung

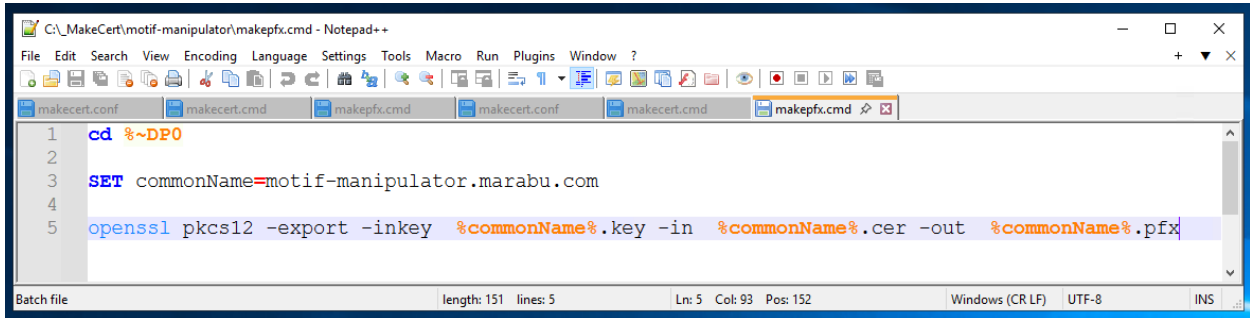


Abbildung 3: Zertifikats-Skript Nr. 2

Nach dem Ausführen der Skripte wird eine .pfx-Datei erstellt. Diese kann in der Web-App nun unter „Certificates“ hochgeladen und ausgewählt werden.

[Abbildung 27: .pfx-Zertifikat hochladen]

3.3.4 Zugriffsregeln über die Firewall

Um das Erstellen der Regeln zu vereinfachen wird die Web-App als Objekt auf der Firewall angelegt. In diesem Objekt wird die private IPv4-Adresse des Private EndPoints hinterlegt. Somit muss in Zukunft nicht die IP-Adresse angegeben werden, sondern nur der Name des Objektes (ähnlich zu Web-App-Name) um dieses in einer Firewall-Regel zu verwenden.

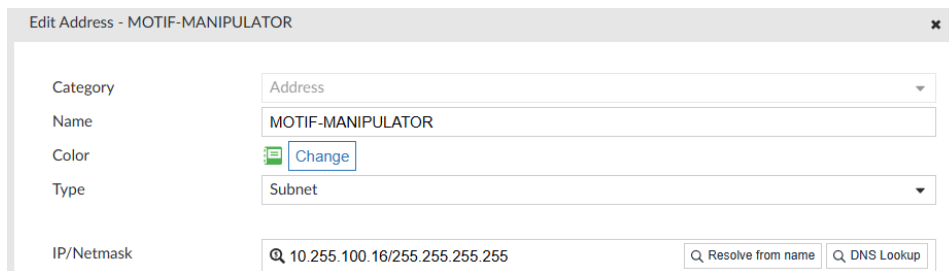


Abbildung 4: Firewall Objekt anlegen

Mit dem angelegten Objekt kann nun die erste Firewall-Regel erstellt werden. Diese ermöglicht den Zugriff der Tochtergesellschaften aus dem LAN-VNet heraus auf die Web-App über das HTTPS-Protokoll. Für jeden Standort gibt es einen VPN-Tunnel zum LAN-VNet in Azure. Dieser VPN-Tunnel wird als „Incoming Interface“, also als Eingang zum Private EndPoint verwendet. Die Tochtergesellschaften werden hier als eine logische Gruppe („Marabu Branches“) zusammengefasst.

Projektdurchführung

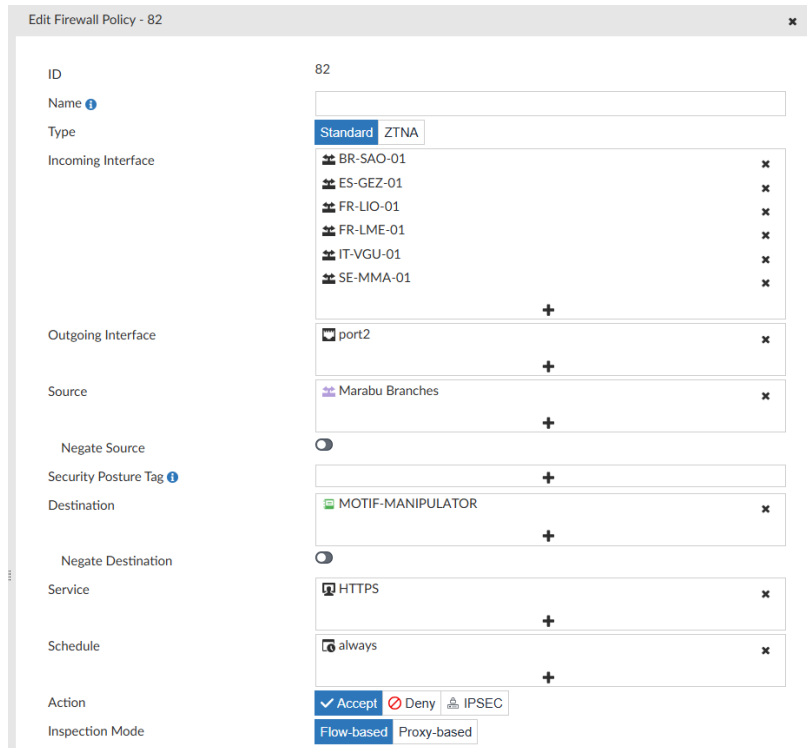


Abbildung 5: Firewall Regel - BranchesToWebApp

Die zweite Regel erfüllt denselben Zweck, nun jedoch für das Hauptnetzwerk hier in Tamm. Hier wird nun der Zugriff für alle betroffenen VLANs in Deutschland ermöglicht.

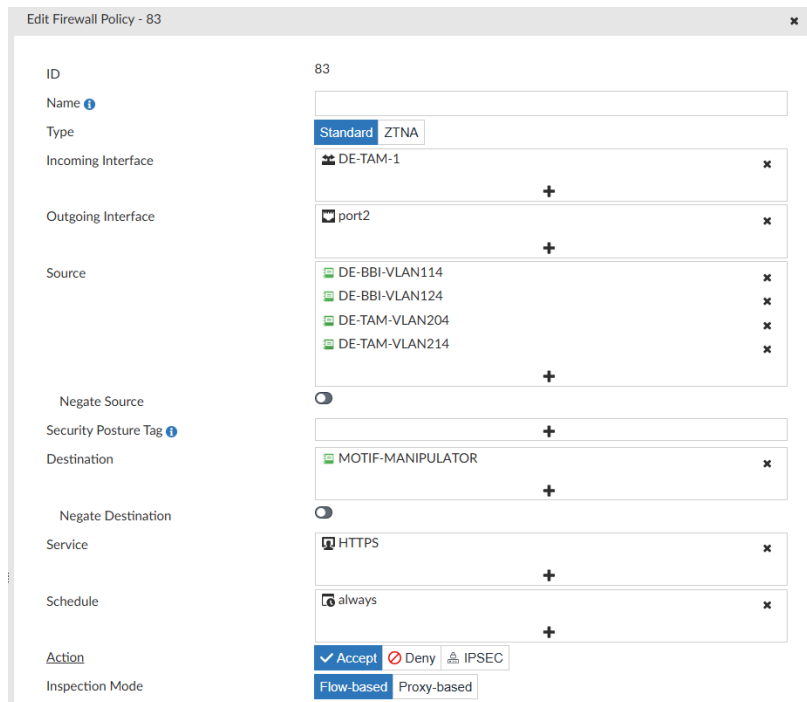


Abbildung 6: Firewall Regel - DEToWebApp

Projektdurchführung

Die Netzwerke der Tochtergesellschaften sind, wie auch hier in Tamm, in unterschiedliche VLANs aufgeteilt. Die VLAN-Aufteilung bleibt von Tochter zu Tochter dieselbe, so ist an jedem Standort das VLAN15 für Clients (z.B. Laptops) zuständig. Damit die Clients der Fachabteilung in Schweden auf das LAN-VNet in Azure zugreifen können, muss eine weitere Regel für alle Client-IP-Adressen erstellt werden.

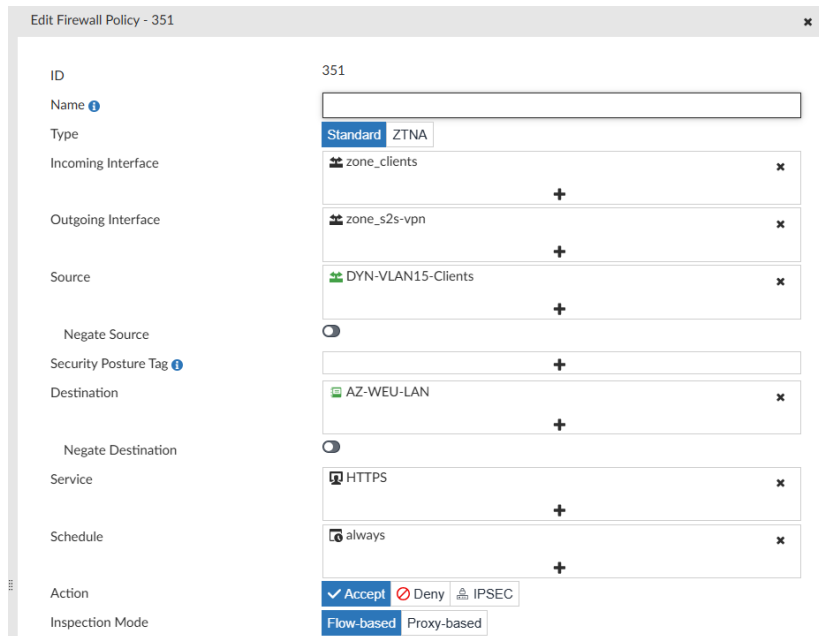


Abbildung 7: Firewall Regel - BranchesClientsToAzure

Mit dieser Regel können nun Clients von jeder Tochtergesellschaft auf das LAN-VNet in Azure zugreifen. Eine tatsächliche Verbindung wird dennoch nicht zustande kommen, da noch keine Route für den Rückweg der Pakete definiert wurde. Azure muss bei Anfragen auch Antworten können, dafür wird nun eine statische Route nach Schweden angelegt. Falls in Zukunft weitere Standorte auf die Web-App zugreifen möchten, müssen hier für jedes Land bzw. jeden Standort weitere Routen angelegt werden.

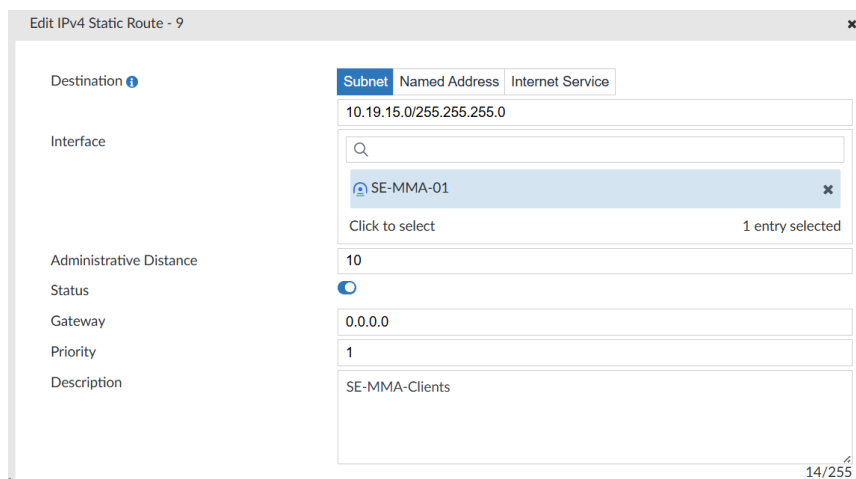


Abbildung 8: Firewall Regel - AzureToClientsSE

Projektdurchführung

3.4 Testphase & Qualitätssicherung

Die Qualität und der Erfolg des Projektes werden durch organisiertes User-Testing sichergestellt. Es werden unterschiedliche Szenarien definiert und durch zwei Parteien durchgeprüft. Auf der einen Seite teste ich, mit dem nötigen technischen Hintergrundwissen, und auf der anderen Seite testet die Fachabteilung, da sie die künftigen User der Web-App sind. Im Folgenden befindet sich eine Tabelle mit den Testszenarien inklusive den Testergebnissen. Während den Tests wird außerdem ein Fokus auf die Skalierbarkeit der verwendeten VM gelegt. Wenn zu viele User (oder zu große Dateien hochgeladen werden) gleichzeitig auf die Web-App zugreifen und diese produktiv nutzen, darf die VM nicht überlastet werden. Es muss abgestimmt werden, wie viele Ressourcen benötigt werden, um einen reibungslosen Betrieb garantieren zu können.

Für die User-Tests werden unterschiedliche Testfälle untersucht. In der folgenden Tabelle sind für jeden Testfall das Thema, das Ergebnis und die darauffolgenden Anpassungen am System angegeben.

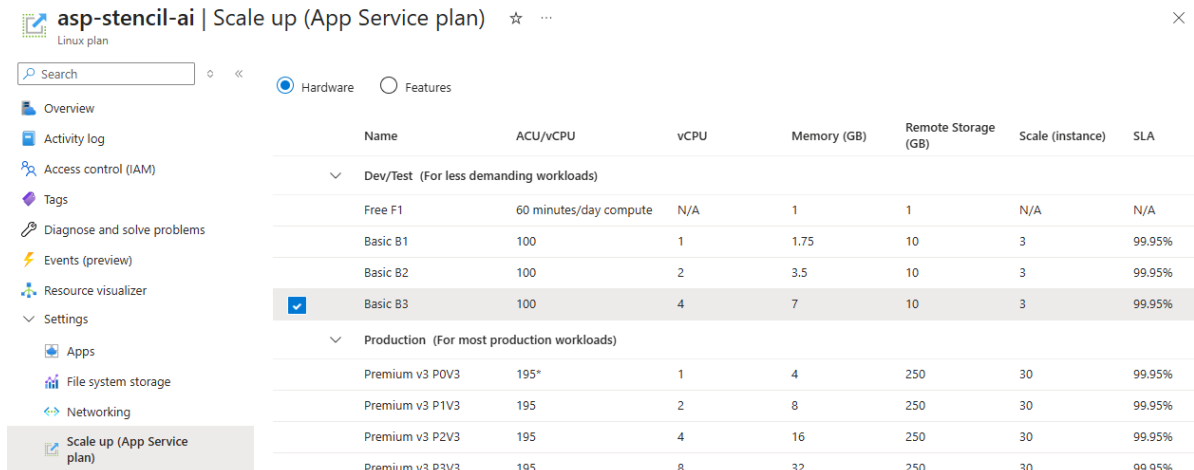
Nr.	Was wird getestet?	Wie wird getestet?	Erwartetes Ergebnis	Tatsächliches Ergebnis	Anpassungen
1	Firewall Regeln, Private EndPoint	Zugriff aus LAN in Schweden (Malmö)	Zugriff ist möglich	Zugriff ist möglich	---
2	Firewall Regeln, Private EndPoint	Zugriff aus LAN in Schweden (Stockholm)	Zugriff wird blockiert	Zugriff wird blockiert	---
3	Firewall Regeln, Private EndPoint	Zugriff aus LAN in Deutschland (Tamm)	Zugriff ist möglich	Zugriff ist möglich	---
4	Firewall Regeln, Private EndPoint	Zugriff aus LAN in Deutschland (BiBi)	Zugriff ist möglich	Zugriff ist blockiert	Statische Route für BiBi anlegen
5	Firewall Regeln, Private EndPoint	Zugriff aus Internet	Zugriff wird blockiert	Zugriff wird blockiert	---
6	VM Skalierung	Verarbeitung großer Dokumente	Leistung der Web-App wird nicht beeinflusst	Web-App ist viel zu langsam	Höheren VM-Preistarif auswählen
7	VM Skalierung	Gleichzeitige Dokumentenverarbeitung	Leistung der Web-App wird nicht beeinflusst	Web-App wird langsamer	Höheren VM-Pricing Plan auswählen

Tabelle 4: Testfall-Matrix

Das vorhandene Problem bei Testfall Nr. 4 (Blockierter Zugriff aus Bietigheim-Bissingen) wird mit Hilfe eines Analyzers gelöst. Hier kann der Traffic der Firewall kontrolliert werden, so wird klar, dass die Pakete vom Client zur Web-App zwar ankommen, die Antwort jedoch nicht durchgestellt wird. Dasselbe Problem trat bereits während der Konfiguration der Firewall-Regeln für Schweden auf, demnach ist dieses Fehlverhalten bereits bekannt und kann umgehend gelöst werden.

Projektdurchführung

Ähnlich ist es bei den Testfällen Nr. 6 und Nr. 7 (Geringe Leistung der Web-App). Bei der Erstellung des ASP (VM) wurde zu Beginn der niedrigste Preistarif ausgewählt, also eine VM mit der geringstmöglichen Leistung. Der Grund für diese Entscheidung ist, dass für das Projekt möglichst viele Kosten gespart und jede Ressource auf die jeweilige Auslastung angepasst werden sollen. Es stellt sich nun heraus, dass ein weitaus höherer Preistarif für die VM ausgewählt werden muss. Anstatt des aktuellen Preisplans B1 wird die höhere Stufe B3 ausgewählt. Hierbei stehen ab sofort 3 weitere virtuelle Kerne (vCPU) sowie 5,25 GB mehr Arbeitsspeicher zur Verfügung. Ein weiterer Durchlauf derselben Testfälle zeigt, dass die vertikale Skalierung erfolgreich ist und die Verwendung der Web-App auch unter größeren Lasten reibungslos funktioniert.



asp-stencil-ai | Scale up (App Service plan) ☆ ...

Linux plan

Search

Hardware Features

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Events (preview)

Resource visualizer

Settings

Apps

File system storage

Networking

Scale up (App Service plan)

Name	ACU/vCPU	vCPU	Memory (GB)	Remote Storage (GB)	Scale (instance)	SLA
Dev/Test (For less demanding workloads)						
Free F1	60 minutes/day compute	N/A	1	1	N/A	N/A
Basic B1	100	1	1.75	10	3	99.95%
Basic B2	100	2	3.5	10	3	99.95%
<input checked="" type="checkbox"/> Basic B3	100	4	7	10	3	99.95%
Production (For most production workloads)						
Premium v3 P0V3	195*	1	4	250	30	99.95%
Premium v3 P1V3	195	2	8	250	30	99.95%
Premium v3 P2V3	195	4	16	250	30	99.95%
Premium v3 P3V3	195	8	32	250	30	99.95%

Abbildung 9: ASP Preistarife

4 Projektabschluss

4.1 Projektrückblick

Das Projekt verlief im Ganzen ohne größere Probleme. Den ursprünglichen Projektplan einzuhalten, der auch im Projektantrag geschildert wurde, erwies sich als nur schwer möglich. Einige der Konzepte wurden während des Projektes aufgrund neuer Erkenntnisse umgestaltet. So musste zum Beispiel kein eigenes VNet mit Subnetzen für die im Projektumfeld angelegte Ressourcengruppe erstellt werden, sondern nur ein einzelner Private EndPoint, welcher mit bereits vorhandenen VNets verbunden wurde. Des Weiteren wurde der VPN-Zugriff des Entwicklers noch nicht eingerichtet. Dieser Punkt ist für den Erfolg dieses Projekt zu vernachlässigen und wird aufgrund fehlender Zeit als offenes „To-Do“ angesehen und nach Projektabschluss nachgeholt. Zusätzlich dazu fällt die Migration der Azure Subscription weg. Ursprünglich war es geplant, die Ressourcengruppe aus der „Stencil AI“ in die „Marabu Main“-Subscription zu migrieren, da dies zuvor der einzige produktive Arbeitsbereich in Azure war. Ich habe mich am Ende aufgrund der Kostenrechnung dagegen entschieden und somit läuft die Web-App nun in einer individuellen Subscription.

Der vor dem Projekt definierte Zeitplan konnte nicht eingehalten werden. Einerseits verringert sich der Aufwand für manche Aufgabenpakete durch eine Neu-Konzeptionierung (wie eben bereits geschildert), andererseits wurde der Aufwand für einzelne Aufgaben unterschätzt und mehr Zeit investiert als ursprünglich geplant. Der tatsächliche Zeitaufwand befindet sich im Anhang [Tabelle 6: Realer Zeitplan].

4.2 Projektaussicht

Nach Abschluss des Projektes wird die Zufriedenheit der Fachabteilung mit der Web-App in einer dreiwöchigen Arbeitsphase bewertet. Je nach Feedback wird ein Nachfolge-Projekt eingeleitet, in dem es um die detailliertere Ausarbeitung des Projektes geht. Bisher ist die Web-App über ein vordefiniertes Passwort geschützt, dieses soll in Zukunft jedoch durch die Verwendung von EntraID abgelöst werden. Außerdem gibt es den Anwendungsfall der Fachabteilung nicht nur in Schweden, sondern bei verschiedenen Tochtergesellschaften mit unterschiedlichen Kunden. Die aktuelle Web-App ist auf den spezifischen Kunden MAPA trainiert und erkennt demnach nur Dokumente, in diesem einen Format. Jeder Kunde verwendet jedoch ein anderes Layout in den Dokumenten, das bedeutet, dass die KI in Zukunft verallgemeinert werden muss, um selbstständig Veränderungen in den Dokumenten wahrzunehmen und zu verstehen.

4.3 Dokumentationen

Die Kundendokumentation bezieht sich in diesem Projekt auf die Fachabteilung in Schweden. Sie wird eine Anleitung beinhalten, in der die Verwendung der Web-App Schritt-für-Schritt beschrieben wird. Die in diesem Projekt verwendete Anleitung ist eine zensierte Version, da das im Original dargestellte PDF-Dokument durch einen NDA-Vertrag geschützt wird und deshalb nicht veröffentlicht werden darf. Neben dieser Anleitung werde ich zusätzlich noch eine zweite Dokumentation anfertigen. Diese Dokumentation wird ein Maintenance-Guide und richtet sich an meine Kollegen in der IT-Abteilung. Das Ziel dieses Guides ist es, das System durchgehend in Betrieb zu halten. So werden beispielsweise das Ablaufen bestimmter Dienste, sowie ein generelles Troubleshooten thematisiert. Beide Dokumentationen befinden sich angehängt als eigenständiges PDF-Dokument.

5 Anhang

5.1 Zeitplanung

Projektphase	Dauer
Anforderungsanalyse	4h
1. Ist/ Soll-Analyse	1h
2. Analyse der bestehenden Infrastruktur	2h
3. Zugriffskonzept	1h
Wirtschaftlichkeitsrechnung	2h
1. Return on Investment	1h
2. Amortisationsrechnung	1h
Projektplanung	4h
1. Ressourcenplanung	1h
2. Planung der Netzwerkanbindung	2h
3. Abstimmung mit IT-Teamleiter	1h
Technische Implementation	15h
1. DevOps einrichten	1h
2. Aufgabenpakete definieren	1h
3. Azure Ressourcen einrichten (+ Migration)	5h
4. Anbindung der Web-App über Private EndPoint	2h
5. Custom Domain mit Zertifikat	2h
6. Firewall Regeln einrichten	4h
Funktionstest mit Anwender	3h
1. Testfälle definieren	1h
2. Test durchführen und Ergebnisse festhalten	1h
3. Fehlerbehebung	1h
Projektdokumentation	12h
1. Projektdokumentation	9h
2. Kundendokumentation	1h
3. Maintenance-Guide	2h
Gesamt	40h

Tabelle 5: Detaillierte Zeitplanung

Anhang

Projektphase	Dauer
Anforderungsanalyse	3,5h
1. Ist/ Soll-Analyse	1h
2. Analyse der bestehenden Infrastruktur	1,5h
3. Zugriffskonzept	1h
Wirtschaftlichkeitsrechnung	2h
1. Return on Investment	1h
2. Amortisationsrechnung	1h
Projektplanung	3h
1. Ressourcenplanung	1h
2. Planung der Netzwerkanbindung	1h
3. Abstimmung mit IT-Teamleiter	1h
Technische Implementation	14h
1. DevOps einrichten	1h
2. Aufgabenpakete definieren	1h
3. Azure Ressourcen einrichten (+ Migration)	3,5h
4. Anbindung der Web-App über Private EndPoint	2h
5. Custom Domain mit Zertifikat	2h
6. Firewall Regeln einrichten	4,5h
Funktionstest mit Anwender	5h
1. Testfälle definieren	1h
2. Test durchführen und Ergebnisse festhalten	2h
3. Fehlerbehebung	1h
 Projektdokumentation	12h
1. Projektdokumentation	9h
2. Kundendokumentation	1h
3. Maintenance-Guide	2h
Gesamt	39,5h

Tabelle 6: Realer Zeitplan

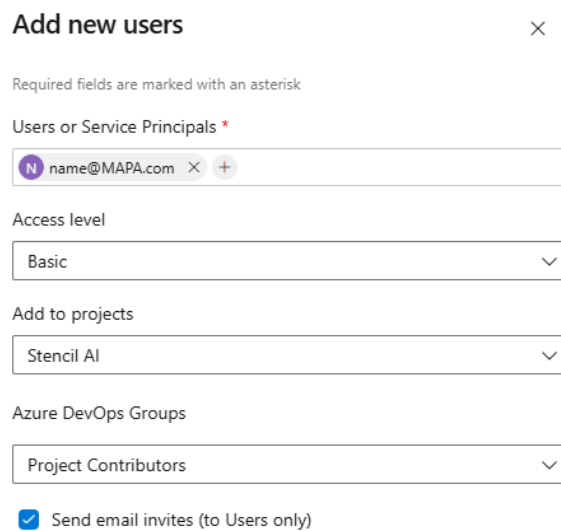
Anhang

5.2 Gantt-Diagramm Arbeitsphasen

ID	Phasen-Bezeichnung	Dauer [in h]	Vorgänger	Nachfolger
A	Ist/ Soll-Analyse	1	-	C, D
B	Analyse der bestehenden Infrastruktur	2	-	G
C	Zugriffskonzept	1	A	G
D	Ressourcenplanung	1	A	E, F
E	Return on Investment	1	D	-
F	Amortisationsrechnung	1	D	-
G	Planung der Netzwerkanbindung	2	B, C	H
H	Abstimmung mit IT-Teamleiter	1	G	I, J, K
I	DevOps einrichten	1	H	-
J	Aufgabenpakete definieren	1	H	K
K	Azure Ressourcen einrichten (+ Migration)	5	H, J	L
L	Anbindung der Web-App über Private EndPoint	2	K	M, N, O
M	Custom Domain mit Zertifikat	2	L	O
N	Firewall Regeln einrichten	4	L	O
O	Testfälle definieren	1	L, M, N	P
P	Test durchführen und Ergebnisse festhalten	1	O	Q
Q	Fehlerbehebung	1	P	R, S, T
R	Projektdokumentation	9	Q	-
S	Kundendokumentation	1	Q	-
T	Maintenance-Guide	2	Q	-

Tabelle 7: Gantt-Diagramm Arbeitsphasen

5.3 DevOps & Azure Screenshots



Add new users ×

Required fields are marked with an asterisk

Users or Service Principals *

name@MAPA.com × +

Access level

Basic ▾

Add to projects

Stencil AI ▾

Azure DevOps Groups

Project Contributors ▾

Send email invites (to Users only)

Abbildung 10: DevOps Userberechtigung

Anhang

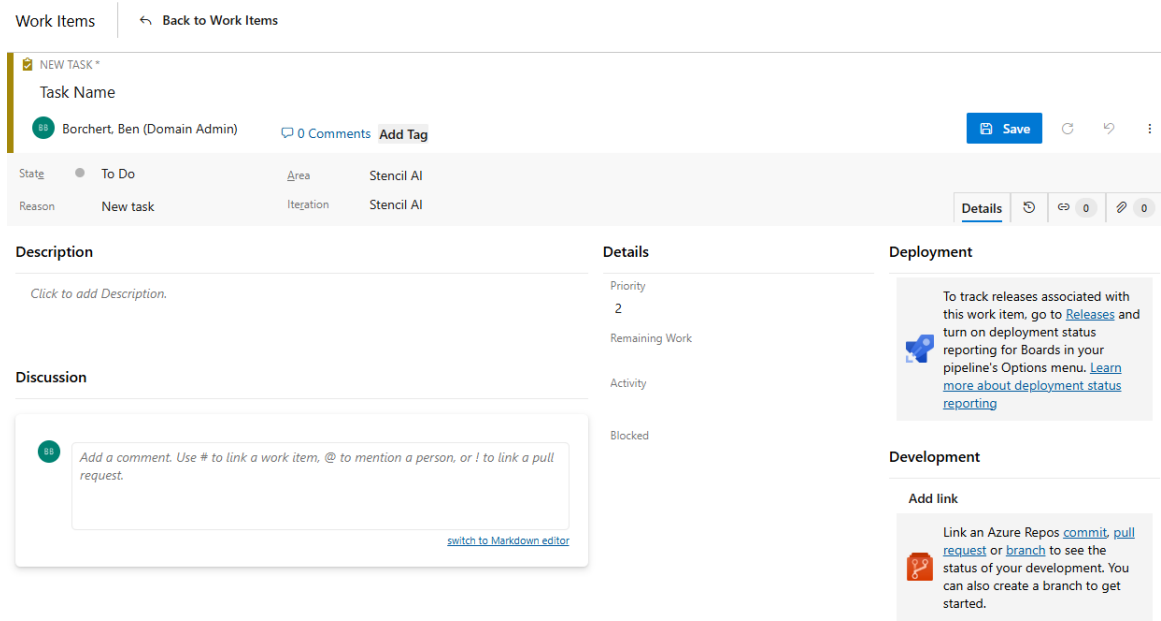


Abbildung 11: DevOps Work-Item

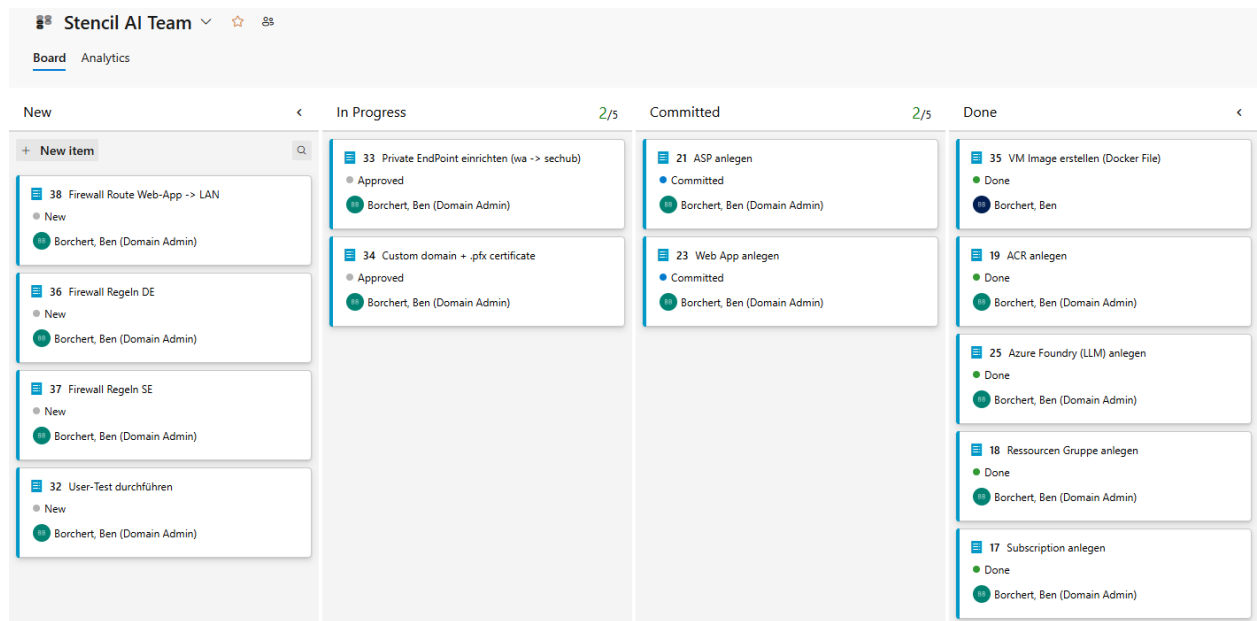


Abbildung 12: DevOps Kanban-Board

Anhang

Create a resource group ...

Basics Tags Review + create


Resource group - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. [Learn more](#)


Subscription * ⓘ Marabu Stencil AI

Resource group name * ⓘ sdc-stencil-ai

Region * ⓘ (Europe) Sweden Central

Abbildung 13: Azure Ressourcengruppe anlegen

Container Registry  ...

 **Container Registry** [Add to Favorites](#)

Microsoft | Azure Service
★ 4.2 (330 ratings)

[Azure benefit eligible](#)

Subscription Marabu Stencil AI Plan Container Registry [Create](#)

Abbildung 14: ACR Ressource

Create container registry ...

Basics Networking Encryption Tags Review + create

Azure Container Registry allows you to build, store, and manage container images and artifacts in a private registry for all types of container deployments. Use Azure container registries with your existing container development and deployment pipelines. Use Azure Container Registry Tasks to build container images in Azure on-demand, or automate builds triggered by source code updates, updates to a container's base image, or timers. [Learn more](#)

Project details

Subscription * Marabu Stencil AI


Resource group * sdc-stencil-ai [Create new](#)

Instance details

Registry name * acrstencilai ✓ .azurecr.io

Location * Sweden Central

Domain name label scope * ⓘ Unsecure

Registry domain name acrstencilai.azurecr.io 

Use availability zones ⓘ Availability zones are activated on premium registries and in regions that support availability zones. [Learn more](#)

Pricing plan * ⓘ Basic

Role assignment permissions mode ⓘ RBAC Registry + ABAC Repository Permissions RBAC Registry Permissions

Abbildung 15: ACR erstellen

Anhang

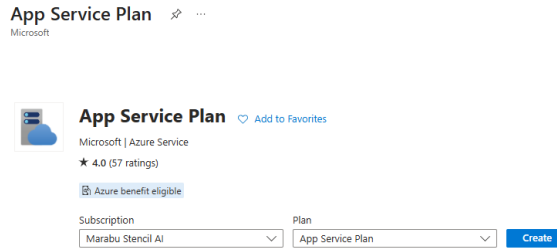


Abbildung 16: ASP Ressource

Create App Service Plan ...

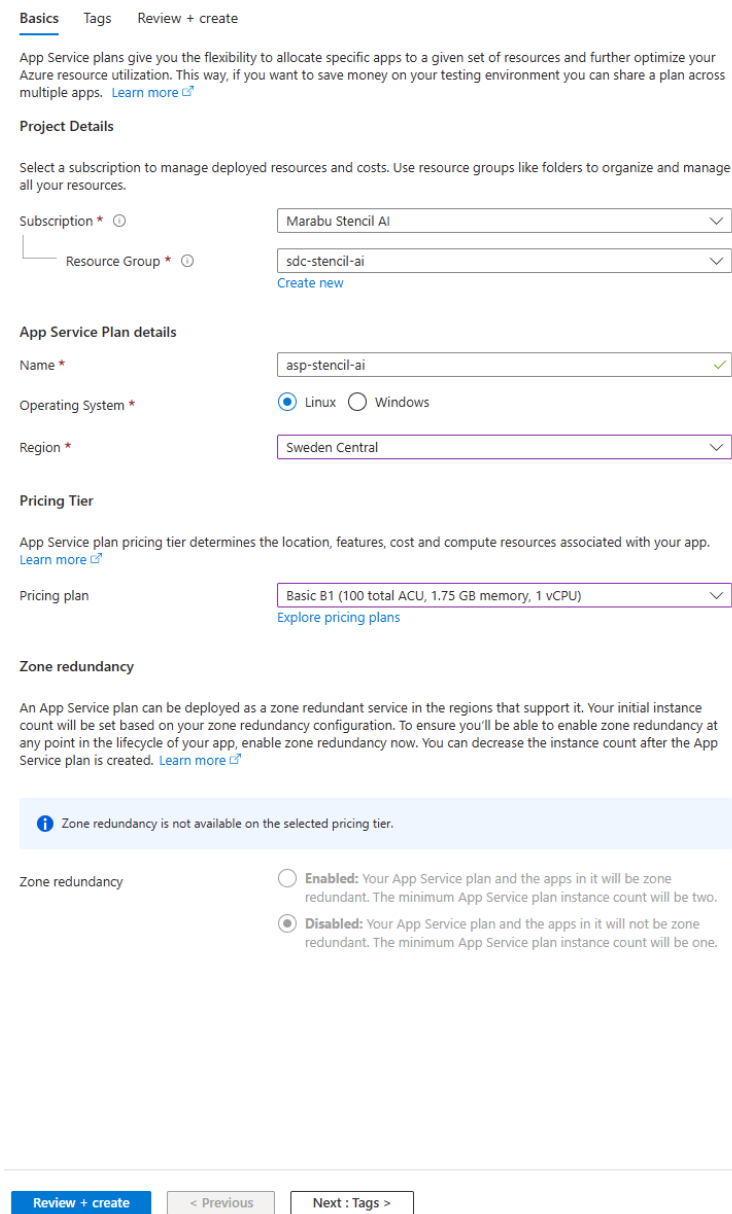


Abbildung 17: ASP anlegen

Anhang

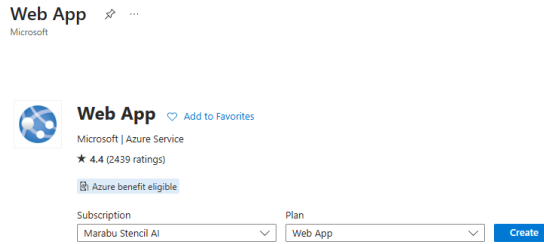


Abbildung 18: Web Service Ressource

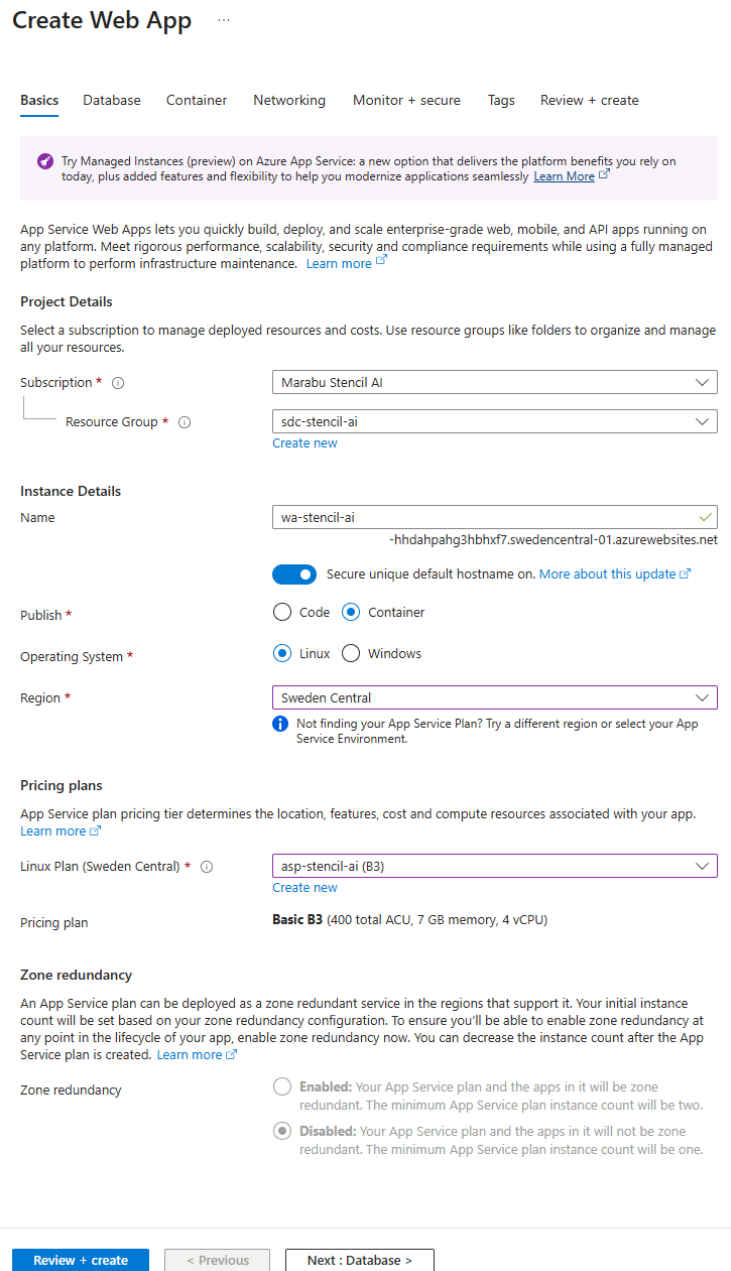


Abbildung 19: Web-App Basic settings

Anhang

Home > sdc-stencil-ai > Marketplace > Web App

Create Web App

Basics Database **Container** Networking Monitor + secure Tags Review + create

Select your preferred source for container images. You can change these settings and other dependencies after creating the app. [Learn more](#)

Sidecar support Enhanced configuration with sidecar support on [Learn More](#)

Image Source * Quickstart Azure Container Registry Other container registries

Name *

Azure container registry options

Registry *

Authentication * Managed identity Admin credentials


Image *

Tag *

Port

Startup Command

Abbildung 20: Web-App Container settings

Microsoft Foundry  [Add to Favorites](#)

Microsoft | Azure Service
★ 5.0 (1 ratings)

Subscription Plan [Create](#)

Abbildung 21: Foundry Ressource

Home > sdc-stencil-ai > Marketplace > Microsoft Foundry

Create a Foundry resource

Basics Storage Network Identity Encryption Tags Review + create

Design, customize, and manage AI apps and agents at scale. [Learn More](#)

Instance Details

Select the subscription to manage deployed resources and costs. Use resources groups like folders to organize and manage all your resources.

Subscription *

Resource group * [Create new](#)

Name *

Region

[View full pricing details](#)

Your first project

Foundry organizes development artifacts into projects— containers for managing permissions, monitoring, costs, and more. When you create a resource, a project is also created.This "default" project has more capabilities than projects that you create later. [Learn more.](#)

Default project name *

Abbildung 22: Foundry anlegen

Anhang

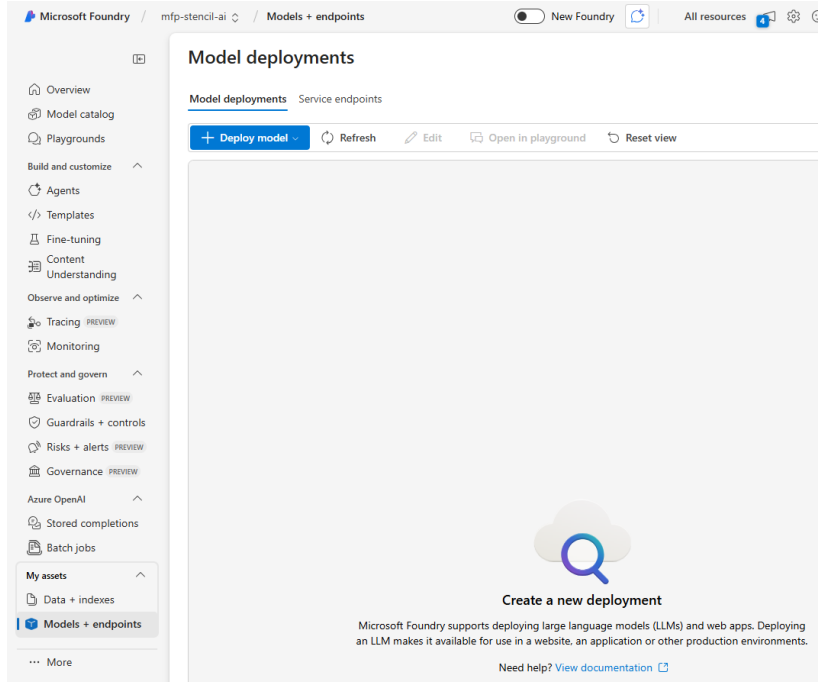


Abbildung 23: LLM model deployment

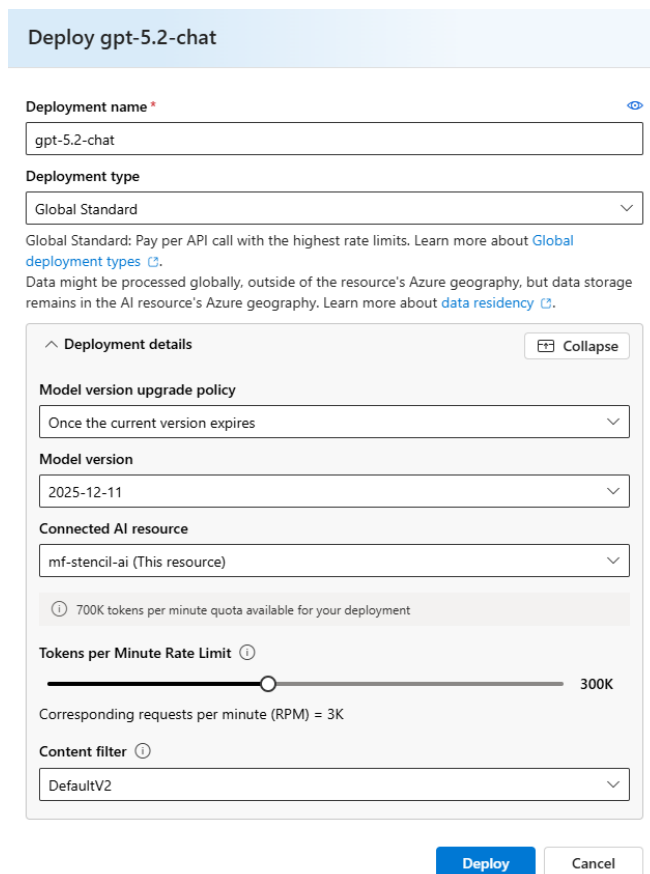


Abbildung 24: LLM deployment Konfiguration

Anhang

Add Private Endpoint ×

Name *

Subscription *

Virtual network *

Subnet *

Integrate with private DNS zone ⓘ
 No Yes

Abbildung 25: Private EndPoint anlegen

Add custom domain ×

Domain provider * ⓘ App Service Domain
 All other domain services

TLS/SSL certificate * ⓘ App Service Managed Certificate
 Add certificate later

Enter your custom domain. We'll give you hostname records to register with your domain provider, and we can validate and add your custom domain here. [Learn more](#) ↗

Domain * ⓘ

Hostname record type * ⓘ

Domain validation

To validate your domain ownership, copy the hostname records below and enter them with your domain provider. If your custom domain uses a load balancer (such as Traffic Manager), make sure the DNS record ultimately resolves to the values listed below. [Learn more](#) ↗

↓ Export to CSV

Type	Host	Value	Status
A			⚠
TXT			✓


⚠ No matching A Record was found. You may still add the custom domain to your web app, but your web app will not receive any traffic until the correct A Record is created.

✓ Validation passed. Select **Add** to finish up.

Abbildung 26: Custom Domain einrichten

Add private key certificate ✕

Source * ⓘ

PFX certificate file * ⓘ
 

Certificate password * ⓘ


Certificate friendly name * ⓘ
 

Abbildung 27: .pfx-Zertifikat hochladen

5.4 Quellenangabe

[Was ist ein privater Endpunkt? - Azure Private Link | Microsoft Learn](#)

[Microsoft Azure \(Cloud\) Solutions Architect - Folge 32 - Azure Private Endpoint - was ist das?](#)

[Was ist ein DNS-TXT-Eintrag? | Cloudflare](#)

[azure - Is a user with "Contributor" and "User Access Administrator" roles same as a user with "Owner" role? - Stack Overflow](#)

<https://docs.fortinet.com/document/fortigate/7.6.4/administration-guide/721410/inspection-modes>

Lehrmaterial aus dem Berufsschulunterricht