

Konzeptpapier: Azure-Sprachbot zur Registrierung von Benutzerdaten

Testat: Advanced Topics in Cloud Computing

Alina Holz

Semester 1

Master Informatik

SoSe 2025

Inhalt

Dialogfluss.....	3
------------------	---

Dialogfluss

Rot: User

Blau: Bot

„Guten Tag, Ich bin ein Azure-Sprachbot, entwickelt zur Erfassung von Benutzerdaten. Ich werde nacheinander für die Erstellung Ihres Nutzeraccounts relevante Daten erfassen. Bitte warten Sie, bis Ich mit dem Sprechen fertig bin, bevor Sie langsam und deutlich antworten. Sie erhalten zum Ende des Erfassungsprozesses noch die Möglichkeit, Ihre Angaben zu ändern oder zu korrigieren.

Bitte sagen Sie mir als erstes jetzt Ihren Vornamen.“

„Max“

„Vielen Dank, sagen Sie mir nun bitte Ihren Nachnamen.“

„Mustermann“

„Vielen Dank, sagen Sie mir bitte als Nächstes Ihre E-Mail-Adresse.“

„Max.mustermann@web.de“

„Vielen Dank, als Nächstes benötige Ich Ihre Telefonnummer“

„1234 5678 90“

„Vielen Dank, als nächstes bitte das Land, in dem Sie wohnen“

„Deutschland“

„Vielen Dank, als nächstes benötige Ich Ihren Wohnort“

„Musterstadt“

„Vielen Dank, als nächstes sagen Sie mir bitte Ihre Postleitzahl“

„12345“

„Vielen Dank, jetzt sagen Sie mir bitte den Namen der Straße, in der Sie wohnen“

„Musterstraße“

„Als nächstes benötige Ich Ihre Hausnummer“

„1“

„Haben Sie einen Adresszusatz?“

„Nein“

„Vielen Dank. Ich wiederhole nun Ihre Angaben. Vorname: Max, Nachname: Mustermann, E-Mail-Adresse: max-mustermann@web.de, Telefonnummer: 1234 5678 90, Land: Deutschland, Wohnort: Musterstadt, Postleitzahl: 12345, Straße: Musterstraße, Hausnummer: 1, Adresszusatz: Keiner. Ist das so korrekt?

„Ja“

„Vielen Dank, damit wurden Ihre Daten erfolgreich erfasst“

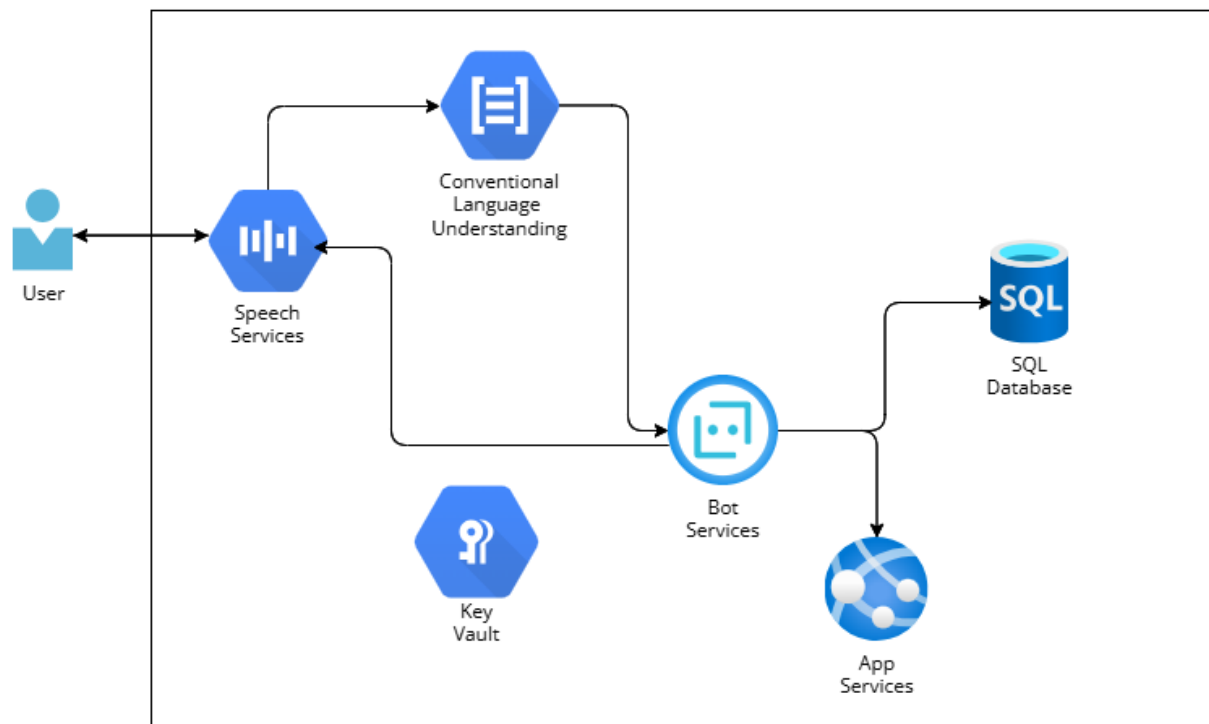
Mögliche Iterationen:

1: Wird der Bot vom User unterbrochen, so wartet dieser, bis der User mit dem Sprechen fertig ist, antwortet: „Bitte warten Sie, bis Ich mit dem Sprechen fertig bin, bevor Sie Ihre Daten übergeben“ und wiederholt die letzte Frage.

2: (Optional) Überprüfung der Angaben des Users. Gibt dieser z.B. bei der Telefonnummer einen Wert an, der keine Telefonnummer ist, so wiederholt der Bot den vom User angegeben Wert, und fragt entweder, ob dieser wirklich korrekt ist, oder lehnt diesen ganz ab. Danach wird die Datenerfassung wiederholt.

3. Antwortet der User auf die letzte Frage, ob die angegebenen Daten so korrekt sind mit „nein“, so wird als erstes gefragt, welcher Wert genau falsch ist. Nach erneuter Erfassung dessen wird gefragt, ob es noch andere Werte gibt, die falsch sind, solange bis der User mit „nein“ antwortet. Danach werden alle Werte erneut wiederholt und erneut gefragt, ob diese korrekt sind. Dieser Prozess wiederholt sich bis zur finalen Bestätigung des Users.

Azure-Architekturdiagramm



Installationsanleitung Azure-Umgebung:

Schritt 1: Ressourcen anlegen

Folgende Ressourcen einfach im Azure Portal anlegen:

- Key Vault
- SQL Datenbank u. Server (wichtig hierbei: Serverless Datenbank ,sonst zu teuer, Gen5, 1vCore)
- App Service (bei mir: nicht der Free App Service Plan, sondern der günstigste danach)
- Conventional Language Understanding Ressource, Region sollte hierbei West Europe sein, die anderen Ressourcen liegen bei mir in Germany West Central
- Bot Ressource
- Azure Speech Ressource

Schritt 2: Secrets im Key Vault ablegen

Einfach im Key Vault im Azure Portal unter Objekte > Secrets. Benötigt wird der Speech Key zur Speech Ressource und der Key für das CLU-Modell, beides einfach zu finden im Azure Portal in der Detailansicht beider Ressourcen.

Schritt 3: App Service Zugriff auf KeyVault gewähren

Der App Service muss zuvor unter EntraID als Managed Identity registriert werden. Dann kann im Key Vault eine Rollenzuweisung geschehen, die dem App Service lesenden Zugriff ermöglicht

Schritt 3: App Service Zugriff auf Datenbank gewähren

Dazu muss auf dem Datenbankserver unter Security > Networking der Zugriff für Azure-Ressourcen erlaubt werden. Dann muss im Query-Editor der Datenbank mit SQL ein user für den App Service angelegt, und diesem lese – und schreibzugriff gewährt werden.

Schritt 4: CLU-Modell trainieren

Das Modell kann unter der Verwendung der Ressource in Azure Language Studio trainiert werden. Ist das Training komplett, so kann das Modell deployed werden und ist unter seinem jeweiligen Endpunkt erreichbar.

Schritt 5: App Service aufsetzen

Der App Service benötigt alle Umgebungsvariablen aus etwaigen lokalen .env-Dateien, unter environment variables. Weiterhin benötigt werden die APPID, und das dazugehörige App Secret der Bot-Ressource, um diese zu verknüpfen. Diese sind

entweder bei der Bot Ressource selbst, oder unter Microsoft Entra ID zu finden.
Weiterhin muss für den App Service die managed identity aktiviert werden. Zugriff benötigt dieser ebenfalls auf die Bot-Ressource.