Architecture Design Document

UPMC Patient Calls Sentiment Analysis and Summarization Application (CSAS)

# Objective/Goals

* Improve Patient Experience and Improve Patient Care by analyzing how the agent conversation occurs, create sentiment analysis, and summarization of call

# High Level Architecture Diagrams

## Overview Diagram

A blue square with black text

Description automatically generated

## Component Diagram

A diagram of a computer program

Description automatically generated with medium confidence

# Architecture Component Details

|  |  |
| --- | --- |
| **Component** | **Description** |
| Azure Storage Container (A, B) | enable a hierarchical namespace when you create a storage account, for NFS protocol support |
| Speech-batch-client | Transcription of very large numbers of audio files against Azure Cognitive Service Speech containers |
| DAEMON mode continuously wait for any new input files and places them on the dispatch queue |
|  | Mounts NFS directory from storage account |
| Azure-speech-container | Enables use of a subset of the Azure AI Speech service features in your own environment |
| The Speech container send billing information to Azure by using a Speech resource on your Azure account. |
| Transcript-to-language Function | An azure function or DAPR pod which:   1. subscribes to Create Blob event 2. invokes azure-language-container to do sentiment analysis, PHI Redaction, and summarization 3. trasnforms the output for Power BI Reporting 4. Stores the output in azure container storage and azure SQL DB |
| Azure-language-container | Enables use of a subset of the Azure AI Language service features in your own environment |
| The Language container send billing information to Azure by using a Azure AI Language resource on your Azure account. |

# Architecture Design Decisions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Decision and Acts** | **Status** | **Rationale/Options** | **Final Decision** |
| 1 | Deploy containers in AKS or host locally | Open |  |  |
| 2 | What pattern/framework to use to build transcript-to-language Function component | Open |  |  |
| 3 | Should raw transcripts be stored in a SQL DB? |  |  |  |
| 4 | How to store raw transcripts in SQL DB since batch-speech-client doesn’t have this capability | Option | Options:   1. Modify the batch-speech client 2. Create a simple Azure Data Factory pipeline which generates DB rows from azure storage files |  |

## Azure Infrastructure Components

1. Azure Storage Account and Blob Containers
2. AKS
3. Azure Container Registry
4. Azure AI Language Service
5. Azure AI Speech Service
6. Azure SQL Server and Database
7. Azure Event Grid
8. Azure Monitor (optional)

# Key Development Tasks

|  |  |  |
| --- | --- | --- |
| **Components** | **Tasks** | **Resource Assignment** |
| Speech-batch-client | * Setting up config.yaml file * Creation of pod yaml file * Enable Blob Storage CSI Driver on AKS cluster. * Creation of PV and PVC to mount NFS directory * Uploading image to ACR |  |
| Azure-speech-container | * Configuring azure-speech-container helm chart * Installing azure-speech-container helm chart to AKS |  |
| Transcript-to-language Function | * subscribes to Create Blob event * invokes azure-language-container to do sentiment analysis, PHI Redaction, and summarization * trasnforms the output for Power BI Reporting * Stores the output in azure container storage and azure SQL DB |  |
| Azure-language-container | * Configuring azure-speech-container helm chart * Installing azure-speech-container helm chart to AKS |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Foundational Use Case Scnarios

1. Generate transcripts from wav files and store them in SQL DB
2. Perform sentiment analysis on call transcripts
3. Perform summarization on call transcripts
4. Perform PHI redaction on call transcripts
5. Transform summarization and sentiment analysis into a format to enable Power BI reporting