

1. Panoramica dell'architettura Dashboard

1.1 Obiettivi principali

1. Gestione dei contenuti RAG:

- Interfaccia per il caricamento e l'eliminazione di documenti.

2. Analisi e monitoraggio:

- Visualizzazione di metriche di utilizzo del chatbot.
- Monitoraggio delle performance del sistema RAG.

3. Sicurezza e controllo accessi:

- Autenticazione degli amministratori tramite Azure AD.
-

2. Autenticazione e Controllo Accessi

2.1 Azure Active Directory

- **Autenticazione utenti/servizi:**
 - Gli amministratori si autenticano tramite **Azure AD**.

2.2 Managed Identities

- **System-Assigned Managed Identity** per l'App Service della dashboard:
 - L'applicazione ottiene token da Azure AD.
 - Accesso a Key Vault, Cosmos DB, database vettoriale e Function App tramite RBAC.
-

3. Infrastruttura di rete condivisa

3.1 Integrazione con la VNet esistente

- **Subnet dedicata** per i componenti della dashboard:
 - Creazione di una subnet specifica (es. `subnet-dashboard`) all'interno della VNet esistente.

3.2 Integrazione dell'App Service con la VNet

- App Service con piano che supporta l'**Azure VNet Integration**:
 - Connessione alla subnet dedicata.
 - Accesso sicuro alle risorse condivise (Cosmos DB, database vettoriale).

3.3 Private Endpoints

- **Accesso alle risorse condivise**:
 - Utilizzo degli stessi private endpoint configurati per il chatbot.
-

4. App Service per la Dashboard

4.1 Architettura dell'applicazione

- **Hosting**:
 - Piano con supporto VNet e prestazioni adeguate.
- **Frontend**:
 - Interfaccia web responsive per la gestione dei documenti.
 - Dashboard per la visualizzazione di metriche e analytics.
- **Backend**:
 - API RESTful per l'interazione con il frontend.
 - Integrazione con Azure Function App per operazioni asincrone.

4.2 Accesso a servizi esterni

- **Cosmos DB:**
 - Accesso in lettura/scrittura per la gestione dei metadati dei documenti e delle chat.
 - Utilizzo della stringa di connessione archiviata in Key Vault.
- **Database Vettoriale:**
 - Operazioni di indicizzazione e gestione dei vettori.
 - Gestione delle credenziali di accesso tramite Key Vault.
- **Blob Storage:**
 - Archiviazione dei documenti originali.
 - Accesso tramite Managed Identity.

4.3 Sicurezza e monitoraggio

- **Application Insights:**
 - Monitoraggio completo delle performance dell'applicazione.
 - **Deployment slot:**
 - Utilizzo di slot di deployment per aggiornamenti senza interruzioni.
 - Testing in ambiente di staging prima del passaggio in produzione.
-

5. Azure Function App

5.1 Architettura serverless

- **Hosting:**
 - Piano Premium per supporto VNet e prestazioni prevedibili.
 - Scalabilità automatica in base al carico di elaborazione.

- **Funzioni:**
 - elaborazione asincrona dei documenti caricati.
 - estrazione di testo dai documenti
 - suddivisione dei documenti in chunk ottimizzati per RAG.
 - generazione di embedding e indicizzazione nel database vettoriale.
 - aggregazione periodica di dati per report e dashboard.

5.2 Integrazione con altri servizi

- **Blob Storage Trigger:**
 - Attivazione automatica delle funzioni al caricamento di nuovi documenti.

5.3 Sicurezza e monitoraggio

- **VNet Integration:**
 - Integrazione con la stessa VNet del chatbot e della dashboard.
 - Accesso sicuro alle risorse tramite private endpoint.
 - **Managed Identity:**
 - Utilizzo di Managed Identity per l'accesso a tutti i servizi Azure.
 - Eliminazione di credenziali hardcoded nel codice.
 - **Logging e monitoraggio:**
 - Logging dettagliato di tutte le operazioni di elaborazione.
-

6. Archiviazione e gestione documenti

6.1 Azure Blob Storage

- **Sicurezza:**
 - Accesso tramite Managed Identity.
 - Crittografia dei dati at-rest e in-transit.

- Soft delete per protezione da cancellazioni accidentali.

6.2 Metadati in Cosmos DB

- **Schema:**
 - Archiviazione di metadati completi per ogni documento (nome, tipo, data, autore, stato di elaborazione).
 - Riferimenti ai chunk generati e agli embedding nel database vettoriale.
 - **Indicizzazione:**
 - Ottimizzazione degli indici per ricerche rapide sui metadati.
 - Supporto per query complesse.
-

7. Analytics e Reporting

7.1 Raccolta dati

- **Metriche di utilizzo:**
 - Tracciamento delle query al chatbot (volume, tipologia, distribuzione temporale).
 - Analisi delle tematiche più frequenti.

7.2 Visualizzazione e reporting

- **Dashboard interattive:**
 - Grafici e visualizzazioni per le principali metriche.
 - Filtri temporali e per categoria.
-

8. Integrazione con il Chatbot

8.1 Condivisione delle risorse

- **Database condivisi:**

- Accesso condiviso a Cosmos DB e database vettoriale.
 - Separazione logica dei dati tramite partitioning o collection dedicate.
- **Infrastruttura di rete:**
- Utilizzo della stessa VNet con subnet separate.
-

Conclusione

Questo documento delinea l'architettura della dashboard amministrativa per il sistema RAG. La soluzione si integra con l'infrastruttura esistente del chatbot, condividendo risorse chiave come database e rete, ma aggiunge componenti specifici per la gestione dei contenuti e l'analisi delle performance. L'architettura è progettata per garantire sicurezza, scalabilità e facilità di gestione, consentendo agli amministratori di controllare efficacemente il sistema RAG e ottimizzarne le prestazioni nel tempo.