## Informazioni sul documento

| | |
|---|---|
| **Task n.** | **4.3** |
| **Tipo** | **SS** |
| **Titolo** | **REALIZZAZIONE DEL DATAWAREHOUSE** |
| **Deliverable previsto nel progetto** | **DOCUMENTAZIONE INTERNA ED ESTERNA DEL PRODOTTO REALIZZATO** |
| **Partecipante/i** | **NETTUNO SOLUTIONS, ENGINFO CONSULTING SRL** |

## Sommario

Scopo del documento ..... 4
Struttura del documento ..... 4
Acronimi e termini chiave ..... 4
1. Attività già realizzate nel SAL1 ..... 4
2. Implementazione DWH ..... 5
2.1. Schema Entità/Relazione ..... 5
2.2. Procedura di creazione del data warehouse ..... 6
3. Progettazione ed implementazione Infrastruttura ..... 12
3.1. Infrastruttura Cloud ..... 12
3.2. Cloud security ..... 13
3.3. Autenticazione ..... 13
3.4. Autorizzazioni/profilazioni ..... 14
3.5. DBMS ..... 14
3.6. BI users client ..... 14
4. Implementazione ETL (Extract/Transform/Load) ..... 16
4.1. Collegamento server piattaforma CMS ..... 16
4.2. Procedura di primo caricamento ..... 18
4.3. Procedura di aggiornamento ..... 21
5. Progettazione ed implementazione Dashboard ..... 23
5.1. Dashboard Visite/ambienti/opere ..... 27
5.1.1. Obiettivo ..... 27
5.1.2. Descrizione ..... 27
5.2. Dashboard profilo Visitatori/Utenti ..... 28
5.2.1. Obiettivo ..... 28
5.2.2. Descrizione ..... 28
5.3. Dashboard catalogo Opere ..... 29
5.3.1. Obiettivo ..... 29
5.3.2. Descrizione ..... 29
6. Scenari futuri (AI e servizi cognitivi) ..... 30
7. Riferimenti Bibliografici ..... 31

## Indice delle figure

FIGURA 1 : SCHEMA E/R ..... 6
FIGURA 2: SCHEMA LOGICO INFRASTRUTTURA ..... 13
FIGURA 3: GESTIONE SERVER COLLEGATI ..... 17
FIGURA 4: GESTIONE SERVER COLLEGATI – PROPRIETÀ - CONNESSIONE ..... 17
FIGURA 5: GESTIONE SERVER COLLEGATI – PROPRIETÀ - SICUREZZA ..... 18
FIGURA 6: POWERBI DESKTOP - AMBIENTE DI SVILUPPO ..... 24
FIGURA 7: POWERBI DESKTOP – VISUALIZZAZIONE DATI ..... 25
FIGURA 8: POWERBI DESKTOP - QUERY EDITOR ..... 26
FIGURA 9: POWERBI DESKTOP - AMBIENTE DI SVILUPPO (2) ..... 26
FIGURA 10: POWERBI DESKTOP - AMBIENTE DI SVILUPPO (3) ..... 27
FIGURA 11: DASHBOARD VISITE/AMBIENTI/OPERA ..... 28
FIGURA 12: DASHBOARD PROFILO VISITATORI/UTENTI ..... 29
FIGURA 13: DASHBOARD CATALOGO OPERE ..... 30

## Scopo del documento

Il presente documento descrive il lavoro finale svolto per il progetto di data warehousing per un museo smart che si è dotato dell'infrastruttura denominata Museum Smart View, un portale dedicato sia alla pubblicazione online delle opere presenti nel sito museale, sia alla raccolta di informazioni di campo relative al comportamento dei visitatori in merito agli ambienti ed alle opera visitate sia fisicamente che virtualmente sul web.

## Struttura del documento

Il documento è strutturato in 7 sezioni.

## Acronimi e termini chiave

| **ACRONIMO** | **DESCRIZIONE** |
|---|---|
| DWH | Data WareHouse |
| CRM | Customer Relationship Management |
| BI | Business Intelligence |
| E/R | Entità-Relazioni |
| ETL | Extract-Transform-Load |
| DATAMART | |
| Slowly Changing Dimension | |

## 1. Attività già realizzate nel SAL1

Nel task 4.3, coordinato da Nettuno Solutions, è coinvolto il partner ENGINFO.
Le attività svolte sul task allo stato hanno riguardato l'implementazione dello schema del DWH.
Sono state avviate le attività di implementazione delle procedure di ETL.
In particolare, sono state stabilite le modalità di connessione sicura attraverso l'utilizzo di un server collegato tra la piattaforma operazionale e quella di datawarehousing.
Parallelamente sono state analizzate e stabilite le logiche di aggiornamento dei dati. La policy più efficiente e sicura per questo tipo di contesto si è ritenuta essere la cancellazione/reimportazione totale delle tabelle anagrafiche ad ogni aggiornamento, mentre per le altre informazioni si è preferito un meccanismo di aggiornamento giornaliero dei record modificati il giorno precedente.

Inoltre, sono stati fissati definitivamente strumenti e tecnologie implementative anche grazie alla consulenza di M2Team Software che è stata ingaggiata per definire, attraverso un'analisi comparativa, lo strumento di sviluppo per la parte BI e Dashboarding più adatto alle esigenze di progetto.

# 2. Implementazione DWH

## 2.1. Schema Entità/Relazione

Nell'ambito della progettazione dei database, il modello entità-relazione (più comunemente detto modello E-R) è un modello teorico per la rappresentazione concettuale e grafica dei dati ad alto livello di astrazione.

Il modello E-R si basa su un insieme di concetti molto vicini alla realtà di interesse: quindi facilmente intuibili dai progettisti (ed in genere considerati sufficientemente comprensibili e significativi anche per i non-tecnici), ma non implementabili sugli elaboratori. Infatti, pur essendo orientato alla progettazione di basi di dati, il modello prescinde dai criteri specifici di organizzazione fisica dei dati persistenti nei sistemi informatici.

All'interno del modello, le entità rappresentano classi di oggetti (fatti, cose, persone, ...) che hanno proprietà comuni ed esistenza autonoma ai fini dell'applicazione di interesse. Un'occorrenza di un'entità è un oggetto o istanza della classe che l'entità rappresenta. Non si parla qui del valore che identifica l'oggetto ma dell'oggetto stesso. Un'interessante conseguenza di questo fatto è che un'occorrenza di entità ha un'esistenza indipendente dalle proprietà ad essa associate. In questo, il modello E-R presenta una marcata differenza rispetto al modello relazionale nel quale non possiamo rappresentare un oggetto senza conoscere alcune sue proprietà. In uno schema ogni entità ha un nome che la identifica univocamente e viene rappresentata graficamente tramite un rettangolo con il nome dell'entità al suo interno.

Le associazioni (dette anche relazioni) rappresentano un legame tra due o più entità. Il numero di entità legate è indicato dal grado dell'associazione: un buono schema E-R è caratterizzato da una prevalenza di associazioni con grado due. È possibile legare un'entità con sé stessa (attraverso un'associazione ad anello), nonché legare le stesse entità con più associazioni.

Di norma viene rappresentata graficamente da un rombo contenente il nome dell'associazione. Il nome può essere un verbo in modo da fornire una direzione di lettura, oppure può essere un sostantivo in modo da non dare una direzione di lettura. L'orientamento accademico e professionale più recente propende per l'utilizzo del sostantivo proprio per evitare di dare un verso all'associazione.

Le entità e le associazioni possono essere descritte usando una serie di attributi. Tutti gli oggetti della stessa classe entità (o associazione) hanno gli stessi attributi: questo è ciò che si intende quando si parla di oggetti simili. La scelta degli attributi riflette il livello di dettaglio con il quale vogliamo rappresentare le informazioni sulle entità e sulle relazioni. Per ciascuna classe entità o associazione si definisce una chiave. La chiave è un insieme minimale di attributi che identifica univocamente un'istanza di entità o associazione. L'attributo si rappresenta con un'ellisse al cui interno viene specificato il nome dell'attributo o anche semplicemente, nel caso di diagrammi complessi, indicandone solo il nome, eventualmente in corrispondenza. In caso di chiave primaria, il nome dell'attributo viene sottolineato o cerchiato.

Partendo da quanto realizzato nel SAL precedente, si è scelto di creare uno schema E-R secondo le seguenti logiche:

- Tutte le dimensioni degeneri diventano tabelle dimensionali e non attributi delle tabelle dei fatti, in quanto rappresentano dimensioni interessanti condivise tra i data mart.

- Le dimensioni relative alla data sono di tipo normalizzato in formato classico date; gli altri attributi sono rappresentazioni in formato testuale utili ai fini di presentazione nella reportistica.
- Nessuna misura calcolata è presente nella progettazione logica, dato che i valori delle misure derivate non vengono memorizzati nel data warehouse. Esse sono state create direttamente nel tool di Business Intelligence, descritto successivamente.
- Non interessa mantenere lo storico per gli attributi dimensionali che sono stati modificati; pertanto tutte le Slowly Changing Dimension sono gestite con la strategia di Tipo 1, ossia il nuovo valore sostituisce il vecchio valore.

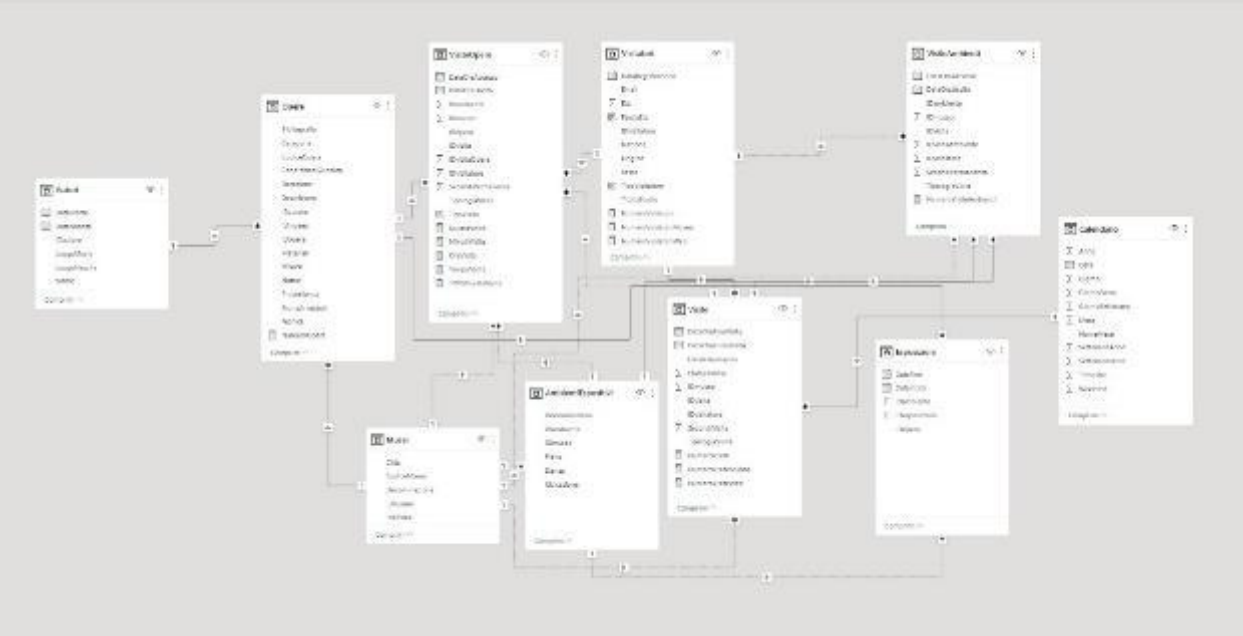

Figura 1 : Schema E/R

## 2.2. Procedura di creazione del data warehouse

La procedura che segue si occupa di creare le tabelle identificate nello schema di cui sopra.

La tabella del calendario è una tabella standard che riporta tutti i dati dei giorni dal 1/1/1970 al 31/12/2099.

```
use [SmartMuseumDWH]

DROP TABLE [dbo].[Calendario]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='Calendario' and xtype='U')
CREATE TABLE [dbo].[Calendario](
        [Data] [date] NULL,
        [Giorno] [int] NULL,
```

```
        [Mese] [int] NULL,
        [Anno] [int] NULL,
        [GiornoAnno] [int] NULL,
        [GiornoSettimana] [int] NULL,
        [Weekend] [int] NULL,
        [Trimestre] [int] NULL,
        [SettimanaAnno] [int] NULL,
        [SettimanaMese] [int] NULL,
        [NomeMese] [varchar](15) NULL
) ON [PRIMARY]

DROP TABLE [dbo].[Musei]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='Musei' and xtype='U')
CREATE TABLE [dbo].[Musei] (
  [IDmuseo] [numeric](11,0) NOT NULL,
  [CodiceMuseo] [nvarchar](255) NULL,
  [Denominazione] [nvarchar](128) NULL,
  [Indirizzo] [nvarchar](100) NULL,
  [Citta] [nvarchar](45) NULL,
  CONSTRAINT [PK_Musei] PRIMARY KEY CLUSTERED
  (
        [IDMuseo] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO

DROP TABLE [dbo].[AmbientiEspositivi]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='AmbientiEspositivi' and xtype='U')
CREATE TABLE [dbo].[AmbientiEspositivi] (
        [IDambiente]  [numeric](11, 0) NOT NULL,
        [IDmuseo] [numeric](11,0) NOT NULL,
        [Denominazione] [nvarchar](128) NOT NULL,
```

```
	[Piano] [nvarchar](1) NULL,
	[Stanza] [nvarchar](45) NULL,
	[Ubicazione] [nvarchar](128) NULL,
	CONSTRAINT [PK_AmbientiEspositivi] PRIMARY KEY CLUSTERED
	(
		[IDambiente] ASC
	)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
	) ON [PRIMARY]
GO

DROP TABLE [dbo].[Esposizioni]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='Esposizioni' and xtype='U')
CREATE TABLE [dbo].[Esposizioni] (
[IDesposizione]  [numeric](11, 0) NOT NULL,
[IDopera] [numeric](11, 0) NOT NULL,
[IDambiente]  [numeric](11, 0) NOT NULL,
[DataInizio] [Date] NOT NULL,
[DataFine]  [Date] NULL,
CONSTRAINT [PK_Esposizioni] PRIMARY KEY CLUSTERED
(
	[IDesposizione] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]


DROP TABLE [dbo].[Opere]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='Opere' and xtype='U')
CREATE TABLE [dbo].[Opere] (
[IDopera] [numeric](11, 0) NOT NULL,
[IDmuseo]  [numeric](11, 0) NULL,
[CodiceOpera] [nvarchar](255) NULL,
```

```
[Nome] [nvarchar](45) NOT NULL,
[Descrizione] text NOT NULL,
[IDautore] [numeric](11) NULL,
[Datazione] [nvarchar](128) NULL,
[Provenienza] [nvarchar](128) NULL,
[Categoria] [nvarchar](128) NULL,
[Materiali] [nvarchar](128) NULL,
[Tecnica] [nvarchar](128) NULL,
[Misure] [nvarchar](128) NULL,
[CondizioneGiuridica] [nvarchar](128) NULL,
[StoriaAneddoti] [nvarchar](max) NULL,
[Bibliografia] [nvarchar](max) NULL,
CONSTRAINT [PK_Opere] PRIMARY KEY CLUSTERED
(
	[IDopera] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

DROP TABLE [dbo].[Visitatori]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='Visitatori' and xtype='U')
CREATE TABLE [dbo].[Visitatori] (
[IDvisitatore] [numeric](11, 0) NOT NULL,
[Email] [nvarchar](512) NOT NULL,
[Sesso] [char](1) NULL, --M/F
[TitoloStudio] [nvarchar](255) NULL,
[DataRegistrazione] [Date] NULL,
[Eta] [numeric](3,0) NULL,
[Origine] [char](1) NOT NULL, --M Museo, W Web
[Nazione] [varchar](2) NULL,
CONSTRAINT [PK_Visitatori] PRIMARY KEY CLUSTERED
(
	[IDvisitatore] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]


DROP TABLE [dbo].[Visite]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='Visite' and xtype='U')
CREATE TABLE [dbo].[Visite] (
[IDvisita] [numeric](11, 0) NOT NULL,
[IDmuseo] [numeric](11, 0) NULL,
[IDvisitatore] [numeric](11, 0) NOT NULL,
[DataRiferimento] [Date] NULL,
[DataOraInizioVisita] [DateTime] NOT NULL,
[DataOraFineVisita] [Datetime] NULL,
[EtaVisitatore] [numeric](3,0) NULL,
[SecondiVisita] [numeric](11,0) NULL,
[TipologiaVisita] [char](1) NOT NULL, --M Museo, W Web
CONSTRAINT [PK_Visite] PRIMARY KEY CLUSTERED
(
	[IDvisita] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]


DROP TABLE [dbo].[VisiteAmbienti]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='VisiteAmbienti' and xtype='U')
CREATE TABLE [dbo].[VisiteAmbienti] (
[IDvisitaAmbiente] [numeric](11, 0) NOT NULL,
[IDvisita] [numeric](11, 0) NOT NULL,
[IDmuseo] [numeric](11, 0) NULL,
[IDambiente] [numeric](11, 0) NOT NULL,
[IDvisitatore] [numeric](11, 0) NOT NULL,
```

```
[DataOraAccesso] [DateTime] NOT NULL,
[DataOraUscita] [Datetime] NULL,
[SecondiPermanenza] [numeric](11,0) NULL,
[TipologiaVisita] [char](1) NULL, --M Museo, W Web
CONSTRAINT [PK_VisiteAmbienti] PRIMARY KEY CLUSTERED
(
	[IDvisitaAmbiente] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]


DROP TABLE [dbo].[VisiteOpere]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='VisiteOpere' and xtype='U')
CREATE TABLE [dbo].[VisiteOpere] (
[IDvisitaOpera] [numeric](11, 0) NOT NULL,
[IDvisita] [numeric](11, 0) NOT NULL,
[IDmuseo] [numeric](11, 0) NULL,
[IDambiente] [numeric](11, 0) NOT NULL,
[IDvisitatore] [numeric](11, 0) NOT NULL,
[IDopera] [numeric](11, 0) NOT NULL,
[DataOraAccesso] [DateTime] NOT NULL,
[DataOraUscita] [Datetime] NULL,
[SecondiPermanenza] [numeric](11,0) NULL,
[TipologiaVisita] [char](1) NULL, --M Museo, W Web
CONSTRAINT [PK_VisiteOpere] PRIMARY KEY CLUSTERED
(
	[IDvisitaOpera] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
```

```
DROP TABLE [dbo].[Autori]
IF NOT EXISTS (SELECT * FROM sysobjects WHERE name='Autori' and xtype='U')
CREATE TABLE [dbo].[Autori] (
[IDautore]  [numeric](11, 0) NOT NULL,
[Nome] [nvarchar](128) NOT NULL,
[DataNascita] [Date] NULL,
[LuogoNascita] [nvarchar](128) NULL,
[DataMorte] [Date] NULL,
[LuogoMorte] [nvarchar](128) NULL
CONSTRAINT [PK_Autori] PRIMARY KEY CLUSTERED
(
        [IDautore] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
```

# 3. Progettazione ed implementazione Infrastruttura

## 3.1. Infrastruttura Cloud

Il prodotto è ospitato su Microsoft Azure, una delle principali piattaforme di cloud computing al mondo, in particolare sul nodo westEurope in Olanda (quindi senza alcun limite per quanto riguarda problematiche di privacy e conservazione dati).
Azure offre un ambiente altamente scalabile e affidabile, consentendo al prodotto di adattarsi alle necessità crescenti degli utenti e garantendo elevati standard di disponibilità e ridondanza.
L'infrastruttura è composta da macchine virtuali, servizi di archiviazione, database e servizi PowerBI configurati in modo da garantire prestazioni ottimali e tempi di risposta rapidi nella massima sicurezza.

Gli utenti (1) accedono all'applicazione previa autenticazione AAD (3) e validazione delle regole firewall (4) per accedere alla rete privata (9). Il server dell'applicazione, un istanza Microsoft IIS utilizza i dati forniti tramite Claims dal precedente blocco di autenticazione AAD (3) quindi, anche interrogando il DB (6) viene deciso il set di dati e i report che l'utente (1) è autorizzato a vedere.
I log prodotti sono archiviati su Storage Azure (8)
Sul sistema sono presenti anche dei Job, sia schedulati che on-demand e che si occupano principalmente della sincronizzazione dei dati e che utilizzano Redis (7) come persistenza.
Alla quasi totalità dei servizi si accede mediante autenticazione integrata, laddove ciò non è possibile, come nel caso di Redis (7), si ricorre a un Azure KeyVault (10) per conservare le credenziali in modo sicuro.

L'applicazione espone inoltre degli endpoints per l'integrazione dei reports in altre applicazioni, mantenendo i medesimi meccanismi di autenticazione/autorizzazione tramite AAD e le misure di sicurezza già in uso durante la fruizione autonoma del servizio.

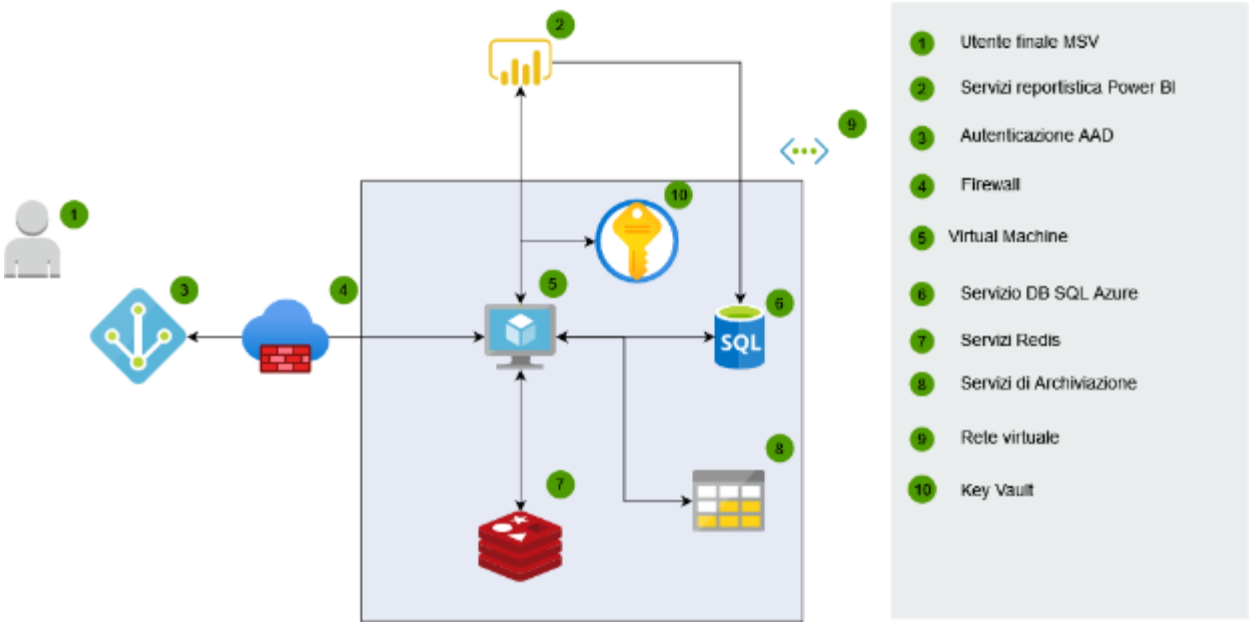


Figura 2: Schema logico infrastruttura

### 3.2. Cloud security

La sicurezza è una priorità fondamentale e Azure offre un robusto modello di sicurezza per proteggere i dati e le risorse del sistema. Vengono adottate pratiche di sicurezza avanzate, come l'autenticazione a più fattori (MFA) per accedere all'applicazione e ai servizi di Azure. L'uso di firewall di rete e regole di rete consentono di limitare l'accesso solo agli indirizzi IP autorizzati, riducendo così il rischio di attacchi da parte di utenti non autorizzati.
L'applicativo è stato sviluppato "secure by default", e nulla è accessibile se non dopo aver assegnato gli opportuni ruoli.
L'uso di un'unica piattaforma cloud ha poi semplificato lo sviluppo e aumentato la sicurezza; l'accesso alle risorse quali il db o i report powerBi non abbisognano di password dedicate che potrebbero divenire pubbliche in caso di incidente, ma ci si avvale piuttosto della sicurezza integrata di Azure, assegnando identità di servizio ai vari attori.

### 3.3. Autenticazione

Il sistema di autenticazione è basato su standard di sicurezza moderni e segue le best practice dell'industria. Gli utenti del prodotto devono autenticarsi attraverso un processo di login sicuro, che richiede credenziali valide, come nome utente e password e un terzo fattore, quale SMS, chiamate telefonica o applicazione di autenticazione su dispositivo mobile personale. Inoltre, il prodotto supporta anche l'autenticazione centralizzata e il Single Sign-On (SSO), semplificando l'accesso agli utenti autorizzati e garantendo un'esperienza utente unificata.

### 3.4. Autorizzazioni/profilazioni

Il progetto implementa un sistema di autorizzazioni e profilazioni avanzato per garantire che gli utenti possano accedere solo alle funzionalità e ai dati appropriati in base al loro ruolo e alle responsabilità assegnate. Vengono definiti ruoli con diversi livelli di autorizzazione, e ogni utente è assegnato a uno o più ruoli in base alle sue necessità. Le autorizzazioni sono gestite in modo granulare, consentendo di limitare l'accesso a specifici rapporti o dati sensibili solo a un gruppo selezionato di utenti.
Es: super amministratore, amministratore di più musei, amministratore di un museo, amministratore di un dipartimento di un museo, etc....

### 3.5. DBMS

Per assolvere al ruolo di archiviazione dati su DB si è scelto SQL Azure. Esso è una versione completamente gestita di SQL Server, progettato specificamente per funzionare su piattaforma cloud. Questo servizio offre prestazioni elevate, scalabilità automatica e funzionalità avanzate di sicurezza e disponibilità dei dati.

Alcune delle motivazioni chiave che hanno guidato la scelta di SQL Azure includono:

- Scalabilità: SQL Azure consente di scalare orizzontalmente e verticalmente i database in modo dinamico in base alle esigenze del carico di lavoro. Ciò è fondamentale per gestire un crescente volume di dati e di utenti senza compromettere le prestazioni nè sprecare risorse con l'acquisto preventivo.
- Affidabilità e Ridondanza: SQL Azure offre un'elevata disponibilità dei dati attraverso repliche anche geograficamente distribuite. I dati vengono automaticamente replicati su più data center, garantendo che in caso di guasto di un centro dati, l'accesso ai dati continui senza interruzioni. E' inoltre disponibile, in modo completamente gestito, il backup dei dati con point in time, con precisione al minuto.
- Sicurezza Avanzata: SQL Azure implementa robuste misure di sicurezza, come la crittografia dei dati in transito e a riposo, consentendo di proteggere le informazioni sensibili degli utenti e garantendo il rispetto delle normative sulla privacy dei dati.
- Monitoraggio e Diagnostica: SQL Azure fornisce strumenti completi per il monitoraggio delle prestazioni del database e l'individuazione di eventuali problematiche. Ciò permette al nostro team di sviluppo di rilevare tempestivamente e risolvere eventuali inefficienze o anomalie nel sistema.
- Integrazione con Power BI e Azure Services: Essendo entrambi prodotti di Microsoft, SQL Azure si integra perfettamente con Power BI e altri servizi di Azure, semplificando lo scambio di dati e la creazione di report interattivi e intuitivi.

### 3.6. BI users client

Power BI è un servizio d'analisi aziendale prodotto da Microsoft. Fornisce visualizzazioni di dati interattive e funzionalità di business intelligence con un'interfaccia grafica, per consentire agli utenti di creare report e dashboard.

Power BI è stato distribuito per la prima volta il 24 luglio 2015.

Nel febbraio 2019, Gartner conferma Microsoft come leader nel "Quadrante magico Gartner 2019 per piattaforma di analisi e business intelligence" per le funzionalità della piattaforma Power BI. Ciò voleva dire che era il 12º anno consecutivo di riconoscimento di Microsoft come fornitore leader in questa categoria di quadranti magici (3 anni prima della creazione di questo strumento).

I componenti chiave dell'ecosistema Power BI comprendono:

Desktop Power BI

L'applicazione basata su desktop Windows per PC e desktop, principalmente per la progettazione e la pubblicazione di report per il Servizio.

Servizio PowerBI

Il servizio online basato su SaaS (software come servizio). Questo era precedentemente noto come Power BI per Office 365, ora denominato PowerBI.com o semplicemente Power BI.

App per dispositivi mobili Power BI

Le app Power BI per dispositivi mobili per dispositivi Android e iOS, nonché per telefoni e tablet Windows.

Gateway di Power BI

I gateway vengono usati per sincronizzare i dati esterni in entrata e in uscita da Power BI e sono necessari per gli aggiornamenti automatici. In modalità Enterprise, può essere utilizzato anche da Power Automate (precedentemente denominato Flows) e PowerApps in Office 365.

Power BI integrato

L'API REST di Power BI può essere usata per creare dashboard e report nelle applicazioni personalizzate che servono gli utenti di Power BI e gli utenti non Power BI.

Server di report di Power BI

Una soluzione di reporting di Power BI locale per le aziende che non archiviano o non possono archiviare i dati nel servizio Power BI basato su cloud.

Power BI Premium

Offerta basata sulla capacità che include la flessibilità di pubblicare report su vasta scala in tutta l'azienda, senza richiedere ai destinatari di ottenere licenze individuali per utente. Scalabilità e prestazioni maggiori rispetto alla capacità condivisa nel servizio Power BI

Marketplace di elementi visivi di Power BI

Un marketplace di elementi visivi personalizzati e elementi visivi basati su R. Puoi dare vita ai tuoi dati aziendali e trarre informazioni da questo risultato.

Flusso di dati di Power BI

Un'implementazione di Power Query nel cloud che può essere utilizzata per le trasformazioni dei dati per creare un set di dati Power BI comune che può essere reso disponibile per diversi sviluppatori di report tramite Common Data Service di Microsoft. Può essere utilizzato come alternativa, ad esempio, all'esecuzione di trasformazioni in SSAS e può garantire che diversi sviluppatori di report utilizzino dati che sono stati trasformati in modo simile.

Set di dati di Power BI

Un set di dati di Power BI può funzionare come una raccolta di dati da utilizzare nei report di Power BI e può essere connesso o importato in un report di Power BI. Un set di dati può essere connesso e ottenere i dati di origine tramite uno o più flussi di dati.

Datamart di Power BI

All'interno di Power BI, Datamart è un contenitore che combina flussi di dati Power BI, set di dati e un tipo di data mart o data warehouse (sotto forma di un database SQL di Azure) nella stessa interfaccia. L'interfaccia ha quindi la possibilità di essere un unico luogo per l'amministrazione sia del livello ETL (Dataflow), un data mart intermediario (con ad esempio l'archiviazione di schemi a stella, tabelle dimensionali, tabelle dei fatti), e infine il livello di modellazione (Dataset).

Hub dati di Power BI

Un hub di dati per l'individuazione dei set di dati di Power BI all'interno del servizio Power BI di un'organizzazione in modo che i set di dati possano essere riutilizzati da una posizione centrale.

## 4. Implementazione ETL (Extract/Transform/Load)

Una volta compresa la struttura della base di dati per il supporto alle decisioni da realizzare si può passare a realizzare il data warehouse e la business intelligence.

### 4.1. Collegamento server piattaforma CMS

La piattaforma CMS si basa sul RDBMS My SQL 8.

Al fine di consentire al motore di database di SQL Server, scelto come rdbms per il data warehouse, di eseguire un'istruzione Transact-SQL che include tabelle in un altro prodotto database come My SQL, è necessario creare quello che in termini tecnici viene definito server collegato (linked server).

Usando Sql Server Management Studio (SSMS), possiamo accedere alla gestione dei server collegati attraverso i nodi Oggetti server (Server Objects) / Server collegati (Linked Server):

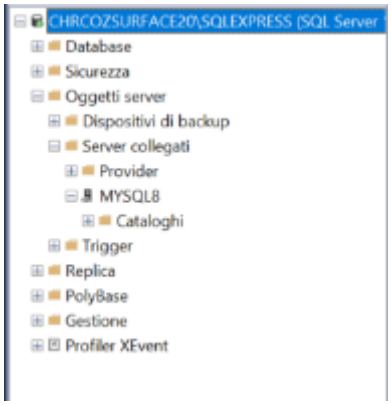


Figura 3: Gestione server collegati

Nel caso specifico si è scelto di creare un Server Collegato ad altra origine dati, in particolare ad una connessione Microsoft OLE DB Provider for ODBC Drivers:

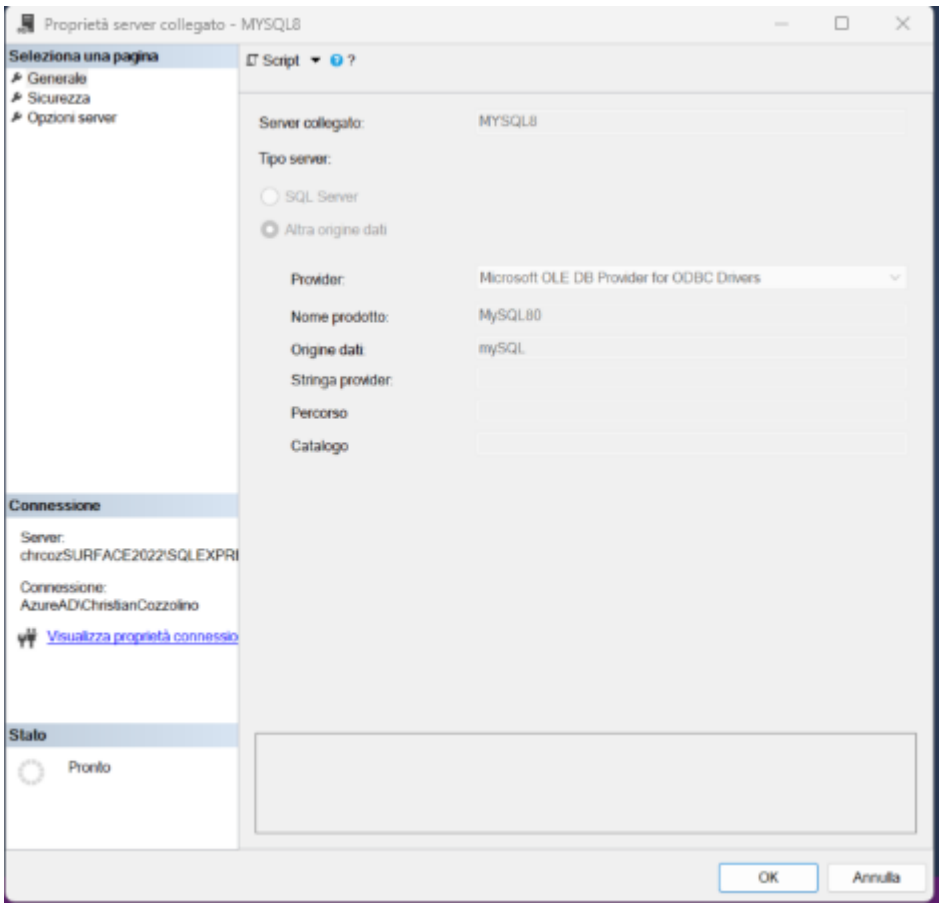


Figura 4: Gestione server collegati – Proprietà - Connessione

Nel tab Sicurezza, si è scelto di effettuare di connettersi usando le credenziali Account di accesso remoto (Remote login) e Password (With password) indicate sotto; ovviamente funziona perché sul server collegato My SQL, che prevede una sua gestione delle autenticazioni, si è precedente creato l'utente di cui sotto con relativa password.

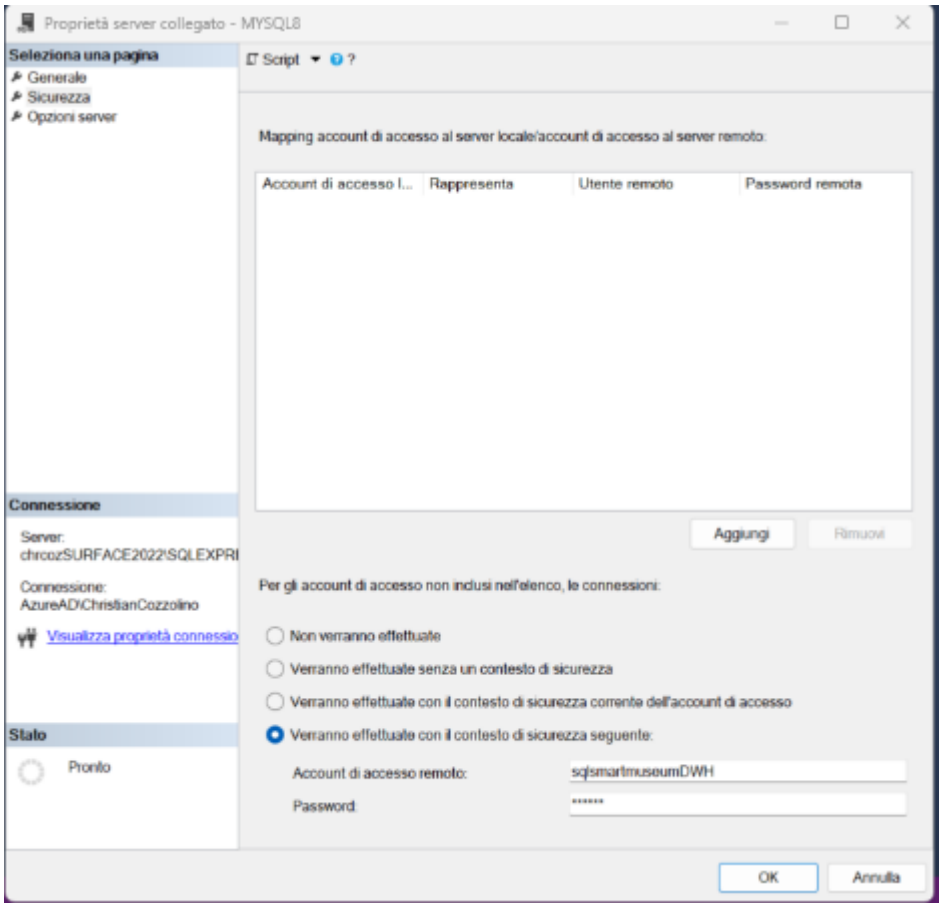


Figura 5: Gestione server collegati – Proprietà - Sicurezza

## 4.2. Procedura di primo caricamento

Al fine di permettere un caricamento iniziale dal CMS, si è sviluppata una procedura di caricamento iniziale che legge tutti i dati dal CMS e li importa nel DWH, secondo le tabelle sviluppate:

```
-- procedura di primo caricamento


-- caricamento totale Musei

Delete From Musei

INSERT INTO Musei

Select * from [mysql8].[museumsmartview]..[msv_spes]
```

```
-- caricamento totale AmbientiEspositivi

Delete From AmbientiEspositivi

INSERT INTO AmbientiEspositivi

Select * from [mysql8].[museumsmartview]..[msv_ambienti_espositivi]


-- caricamento totale Autori

Delete From Autori

Insert INTO Autori

Select * from [mysql8].[museumsmartview]..[msv_opere_autori]


-- caricamento totale Opere

Delete From Opere

Insert INTO Opere

Select id_opera, id_spes, codice_opera, nome, opere.descrizione, id_autore, datazioni.descrizione, provenienze.descrizione, categorie.descrizione,

materiali, tecnica, misure, condizione_giuridica, storia_aneddoti, bibliografia

from [mysql8].[museumsmartview]..[msv_opere] opere

left outer join [mysql8].[museumsmartview]..[msv_opere_datazioni] datazioni on datazioni.id_datazione=opere.id_datazione

left outer join [mysql8].[museumsmartview]..[msv_opere_provenienze] provenienze on provenienze.id_provenienza=opere.id_provenienza

left outer join [mysql8].[museumsmartview]..[msv_opere_categorie] categorie on categorie.id_categoria=opere.id_categoria


-- caricamento totale Esposizioni

Delete From Esposizioni

INSERT INTO Esposizioni

Select id_esposizione, id_Opera, id_ambiente, data_inizio, data_fine from [mysql8].[museumsmartview]..[msv_esposizioni]


-- caricamento totale Visitatori

Delete From Visitatori

INSERT INTO Visitatori
```

```
Select id_visitatore, email, sesso, titolo_studio, data_registrazione, eta,

CASE WHEN origine='Museo' THEN 'M' ELSE 'W' END, nazione from [mysql8].[museumsmartview]..[msv_visitatori]


-- caricamento totale visite

Delete From Visite

INSERT INTO Visite

SELECT id_visita, id_spes, id_visitatore, data_riferimento, data_ora_inizio_visita, data_ora_fine_visita, eta, secondi_visita,

CASE WHEN visita_web='N' THEN 'M' ELSE 'W' END

from [mysql8].[museumsmartview]..[msv_visite]


-- caricamento totale VisiteAmbienti

Delete From VisiteAmbienti where DataOraAccesso

INSERT INTO VisiteAmbienti

SELECT id, va.id_visita, va.id_spes,id_ambiente, vis.id_visitatore, data_ora_accesso, data_ora_uscita, secondi_permanenza,

CASE WHEN vis.visita_web='N' THEN 'M' ELSE 'W' END

from [mysql8].[museumsmartview]..[msv_visite_ambienti] va

inner join [mysql8].[museumsmartview]..[msv_visite] vis

on va.id_visita=vis.id_visita


-- caricamento totale VisiteOpere

Delete From VisiteOpere

INSERT INTO VisiteOpere

SELECT id_visite_opere, vo.id_visita, vo.id_spes,id_ambiente, vis.id_visitatore, id_opera, data_ora_accesso, data_ora_uscita,

secondi_permanenza, CASE WHEN vis.visita_web='N' THEN 'M' ELSE 'W' END

from [mysql8].[museumsmartview]..[msv_visite_opere] vo

inner join [mysql8].[museumsmartview]..[msv_visite] vis

on vo.id_visita=vis.id_visita
```

### 4.3. Procedura di aggiornamento

Per le procedure di aggiornamento quotidiano si è scelto, per le sole tabelle delle Visite, VisiteOpere e VisiteAmbienti, un update parametrizzato a -180 giorni. Chiaramente è un valore che può essere nel tempo rivalutato in base alle valutazioni che verranno fatte di volta in volta, museo per museo.
Per tutte le altre, invece, tra gli approcci possibili per l'estrazione si è scelto quella completa rispetto a quella incrementale, data la ridotta dimensione dei volumi di dati esistenti e la semplicità di implementazione. Le tabelle sono quindi alimentate in modalità TRUNCATE/INSERT, ossia ad ogni esecuzione del flusso vengono ripopolate interamente.

```
--procedura di aggiornamento dati

--aggiornamento totale Musei

Delete From Musei

INSERT INTO Musei

Select * from [mysql8].[museumsmartview]..[msv_spes]


--aggiornamento totale AmbientiEspositivi

Delete From AmbientiEspositivi

INSERT INTO AmbientiEspositivi

Select * from [mysql8].[museumsmartview]..[msv_ambienti_espositivi]


--aggiornamento totale Autori

Delete From Autori

Insert INTO Autori

Select * from [mysql8].[museumsmartview]..[msv_opere_autori]


--aggiornamento totale Opere

Delete From Opere

Insert INTO Opere

Select id_opera, id_spes, codice_opera, nome, opere.descrizione, id_autore, datazioni.descrizione, provenienze.descrizione, categorie.descrizione,

materiali, tecnica, misure, condizione_giuridica, storia_aneddoti, bibliografia

from [mysql8].[museumsmartview]..[msv_opere] opere

left outer join [mysql8].[museumsmartview]..[msv_opere_datazioni] datazioni on datazioni.id_datazione=opere.id_datazione
```

```
left outer join [mysql8].[museumsmartview]..[msv_opere_provenienze] provenienze on provenienze.id_provenienza=opere.id_provenienza

left outer join [mysql8].[museumsmartview]..[msv_opere_categorie] categorie on categorie.id_categoria=opere.id_categoria


--aggiornamento totale Esposizioni

Delete From Esposizioni

INSERT INTO Esposizioni

Select id_esposizione, id_Opera, id_ambiente, data_inizio, data_fine from [mysql8].[museumsmartview]..[msv_esposizioni]


--aggiornamento totale Visitatori

Delete From Visitatori

INSERT INTO Visitatori

Select id_visitatore, email, sesso, titolo_studio, data_registrazione, eta,

CASE WHEN origine='Museo' THEN 'M' ELSE 'W' END, nazione from [mysql8].[museumsmartview]..[msv_visitatori]


--aggiornamento incrementale visite

Delete From Visite where DataRiferimento>=CAST(DATEADD(d, -180, GETDATE()) AS DATE)

INSERT INTO Visite

SELECT id_visita, id_spes, id_visitatore, data_riferimento, data_ora_inizio_visita, data_ora_fine_visita, eta, secondi_visita,

CASE WHEN visita_web='N' THEN 'M' ELSE 'W' END

from [mysql8].[museumsmartview]..[msv_visite]

where data_riferimento>=CAST(DATEADD(d, -180, GETDATE()) AS DATE)


--aggiornamento incrementale VisiteAmbienti

Delete From VisiteAmbienti where DataOraAccesso>=CAST(DATEADD(d, -180, GETDATE()) AS DATE)

INSERT INTO VisiteAmbienti

SELECT id, va.id_visita, va.id_spes,id_ambiente, vis.id_visitatore, data_ora_accesso, data_ora_uscita, secondi_permanenza,

CASE WHEN vis.visita_web='N' THEN 'M' ELSE 'W' END

from [mysql8].[museumsmartview]..[msv_visite_ambienti] va
```

```
inner join [mysql8].[museumsmartview]..[msv_visite] vis
on va.id_visita=vis.id_visita
where data_ora_accesso>=CAST(DATEADD(d, -180, GETDATE()) AS DATE)

--aggiornamento incrementale VisiteOpere
Delete From VisiteOpere where DataOraAccesso>=CAST(DATEADD(d, -180, GETDATE()) AS DATE)
INSERT INTO VisiteOpere
SELECT id_visite_opere, vo.id_visita, vo.id_spes,id_ambiente, vis.id_visitatore, id_opera,
data_ora_accesso, data_ora_uscita,
secondi_permanenza, CASE WHEN vis.visita_web='N' THEN 'M' ELSE 'W' END
from [mysql8].[museumsmartview]..[msv_visite_opere] vo
inner join [mysql8].[museumsmartview]..[msv_visite] vis
on vo.id_visita=vis.id_visita
where data_ora_accesso>=CAST(DATEADD(d, -180, GETDATE()) AS DATE)
```

## 5. Progettazione ed implementazione Dashboard

Per la progettazione delle varie dashboard si è scelto di usare PowerBI Desktop.

Si tratta di uno strumento potente, con molte funzionalità, ideale per l'obiettivo del progetto.

In Power BI Desktop esistono tre aree di lavoro principali, o visualizzazioni: visualizzazione Report, visualizzazione Dati e visualizzazione Relazioni.

Power BI Desktop include anche un editor di query, che viene aperto in una finestra separata.

La visualizzazione Report include cinque aree principali:

- La barra multifunzione, che visualizza le attività comuni associate a report e visualizzazioni;
- La visualizzazione Report, o canvas, dove vengono create e disposte le visualizzazioni;
- La scheda Pagine nella parte inferiore, che consente di selezionare o aggiungere una pagina del report;
- Il riquadro Visualizzazioni, in cui si può cambiare visualizzazione, personalizzare i colori o gli assi, applicare filtri, trascinare i campi e altro ancora;
- Il riquadro Campi, in cui gli elementi della query e i filtri possono essere trascinati nella vista Report oppure nell'area Filtri del riquadro Visualizzazioni.

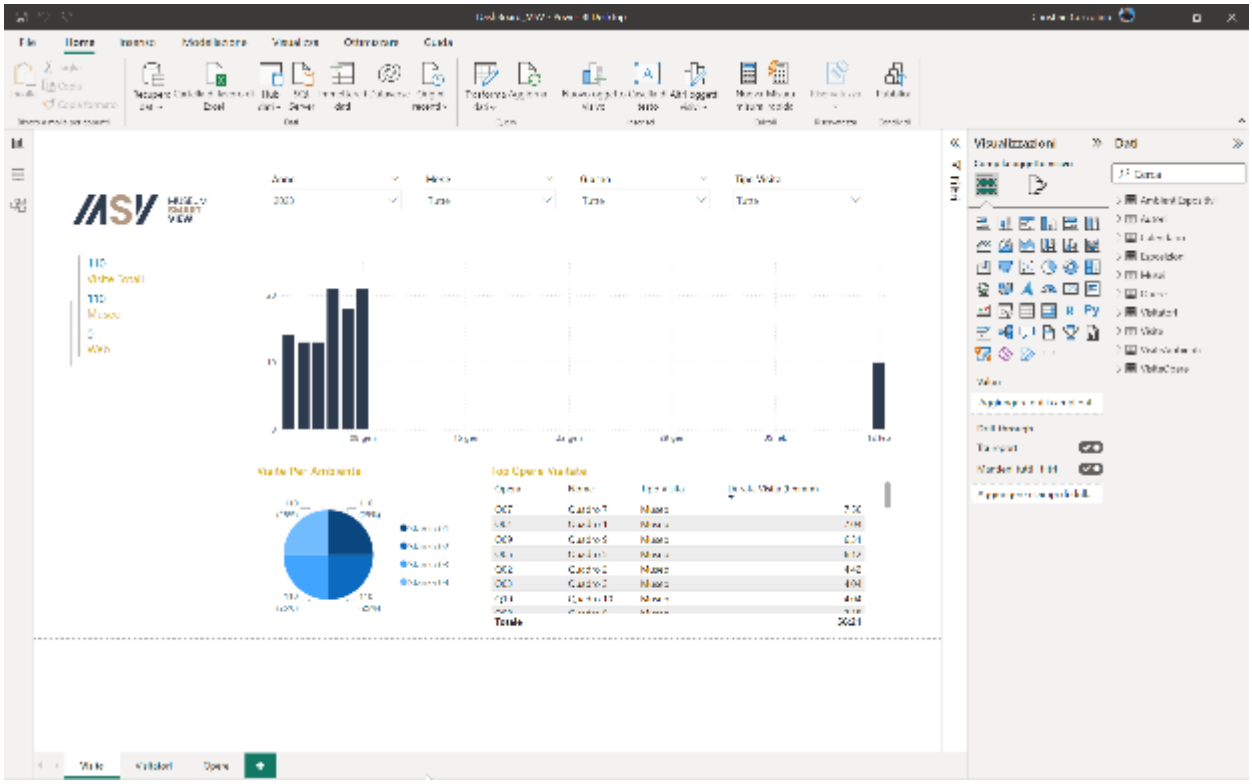

Figura 6: PowerBI Desktop - ambiente di sviluppo

La visualizzazione Dati contiene tre aree principali:

1. La barra multifunzione in cui si possono creare tabelle e colonne calcolate e apportare altre modifiche al modello di dati;
2. Il riquadro centrale, che mostra i dati per la tabella selezionata;
3. Il riquadro Campi, in cui si può controllare la modalità di visualizzazione dei campi nei report.

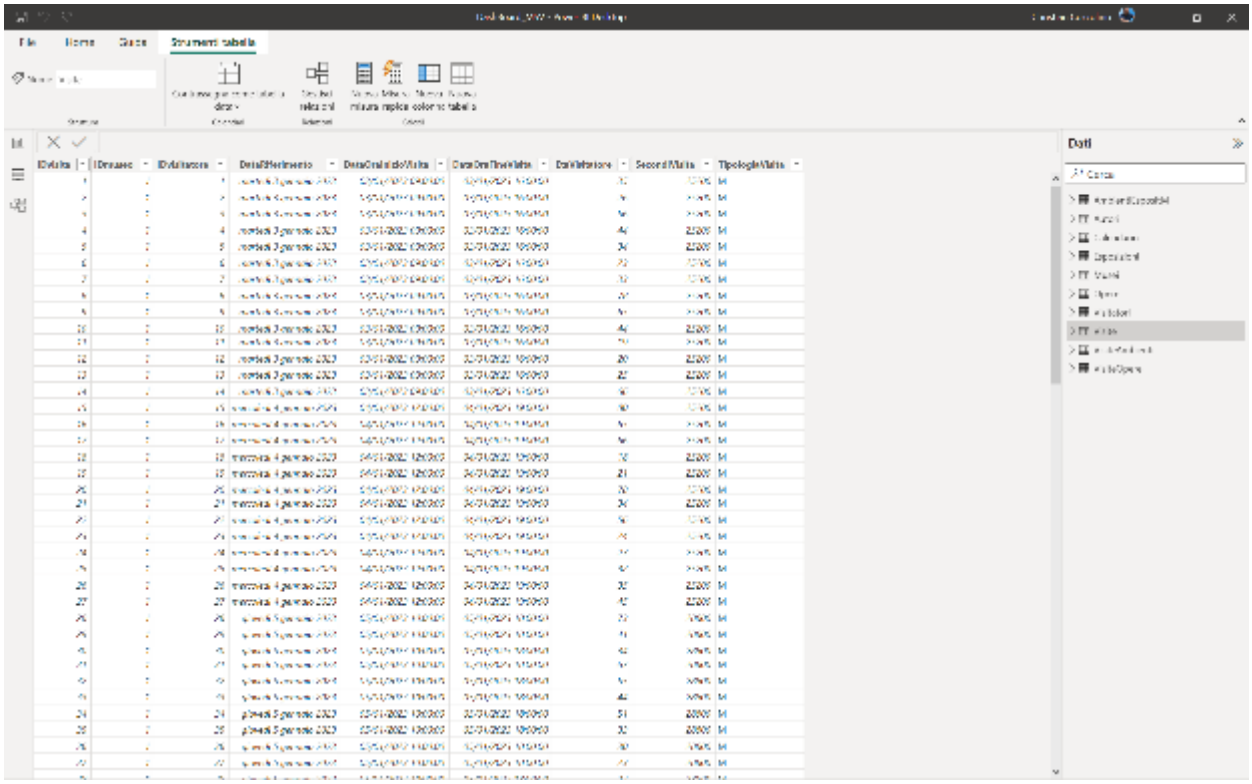

Figura 7: PowerBI Desktop – Visualizzazione dati

Nell'editor di query, è possibile compilare query e trasformare i dati, quindi caricare il modello di dati ottimizzato in Power BI Desktop.

L'editor di query include quattro aree principali:

1. La barra multifunzione, che contiene numerose opzioni per il data shaping e la trasformazione dei dati acquisiti;
2. Il riquadro sinistro, in cui le query vengono elencate e rese disponibili per la selezione, la visualizzazione e il data shaping;
3. Il riquadro centrale, in cui vengono visualizzati i dati dalla query selezionata disponibili per il data shaping;
4. La finestra Impostazioni query, che elenca le proprietà della query e i passaggi di trasformazione dei dati che sono stati applicati.

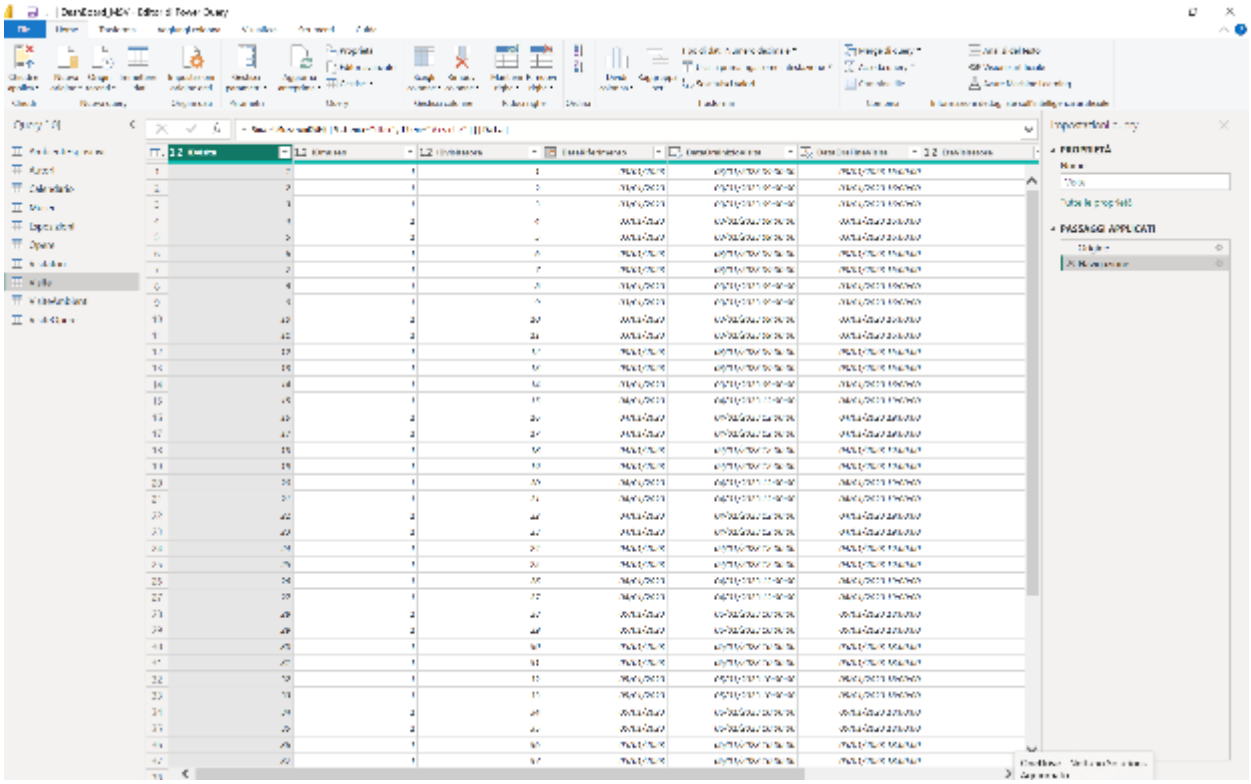

Figura 8: PowerBI Desktop - Query editor

Attraverso questo strumento si è caricata la fonte dati SQL Server del DWH e si sono create le varie misure calcolate di calcolo necessarie per le singole dashboard.

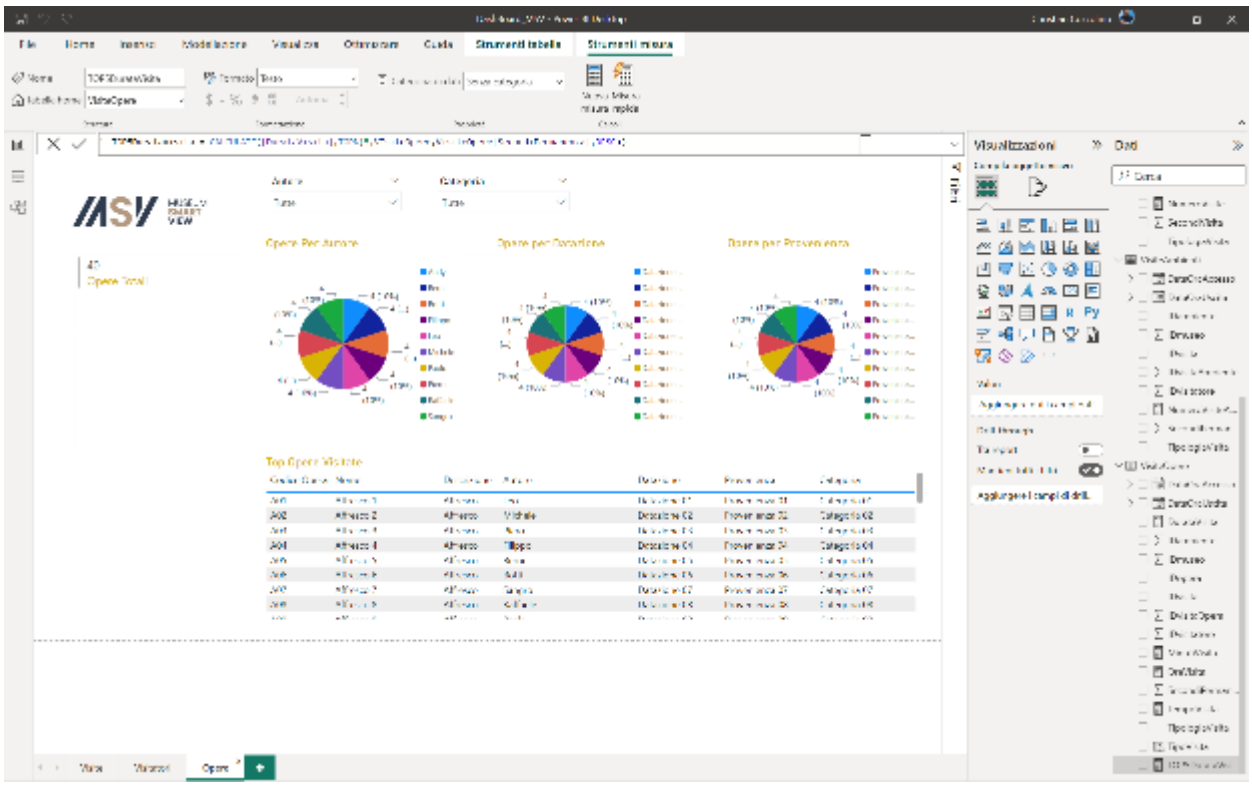

Figura 9: PowerBI Desktop - ambiente di sviluppo (2)

Si è provveduto quindi a gestire le relazioni tra le varie tabelle ed evitare chiaramente i loop per permettere allo strumento di occuparsi delle varie proiezioni in fasi di visualizzazioni dei dati aggregati

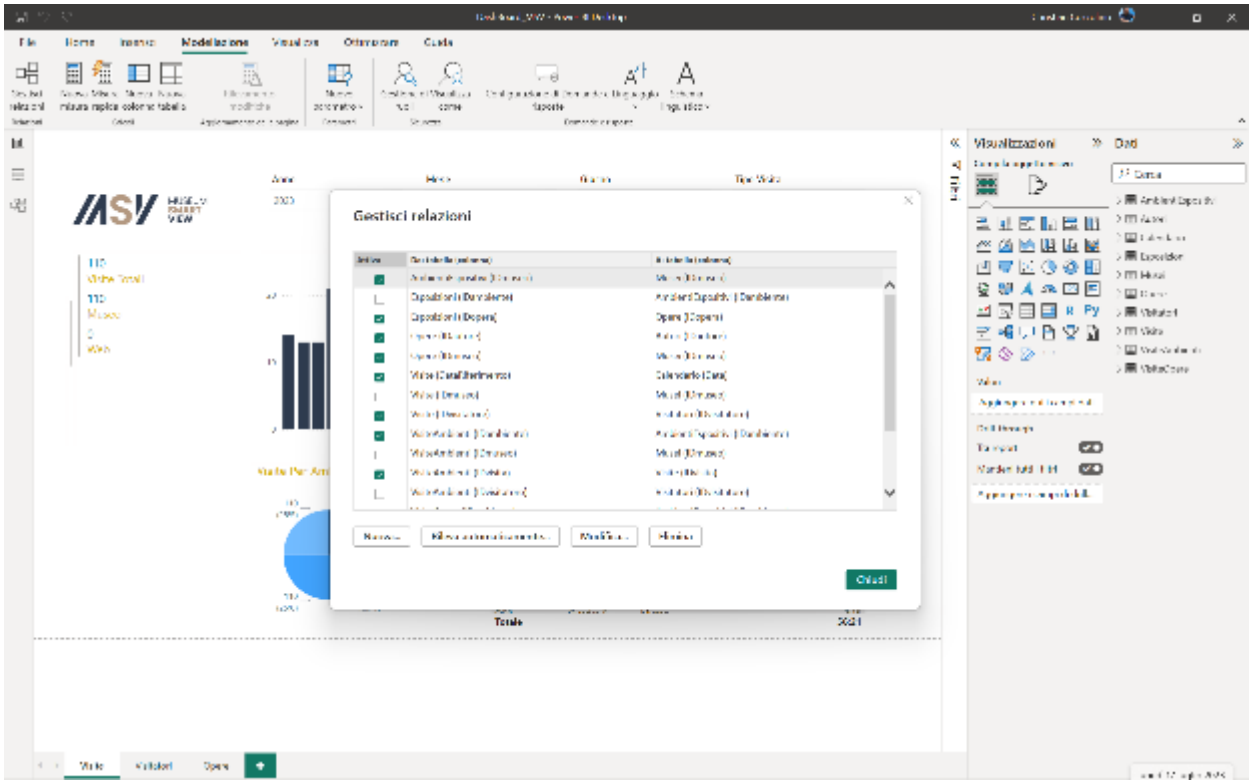


Figura 10: PowerBI Desktop - ambiente di sviluppo (3)

## 5.1. Dashboard Visite/ambienti/opere

### 5.1.1. Obiettivo

La dashboard è stata progettata per dare all'utente amministratore del museo una immediata visione di insieme del comportamento degli utenti/visitatori nel contesto museale.

### 5.1.2. Descrizione

La dashboard permette di gestire filtri sul tempo (anno/mese/giorno) e sulla tipologia di visita (web o presso museo).

A seconda del livello di profilazione dell'utente collegato (amministratore di un singolo dipartimento, amministratore di un intero museo, super amministratore di un'aggregazione di musei) saranno caricati i dati relativi.

Sostanzialmente vengono visualizzati:

- Il totale delle visite e i subtotali per tipologia
- L'andamento giornaliero delle visite nel periodo di interesse
- Il totale delle visite per ambiente espositivo
- Le prime 10 opere più visitate per tipologia di visita

L'utente attraverso l'opzione di drill insita in PowerBI, cliccando su qualsiasi entità grafica della dashboard potrà aggiungere un ulteriore filtro rigenerando al volo la visualizzazione della stessa (es: cliccando su una barra di un giorno dell'istogramma delle visite, la dashboard si ricaricherà con i dati relativi a quello specifico giorno).

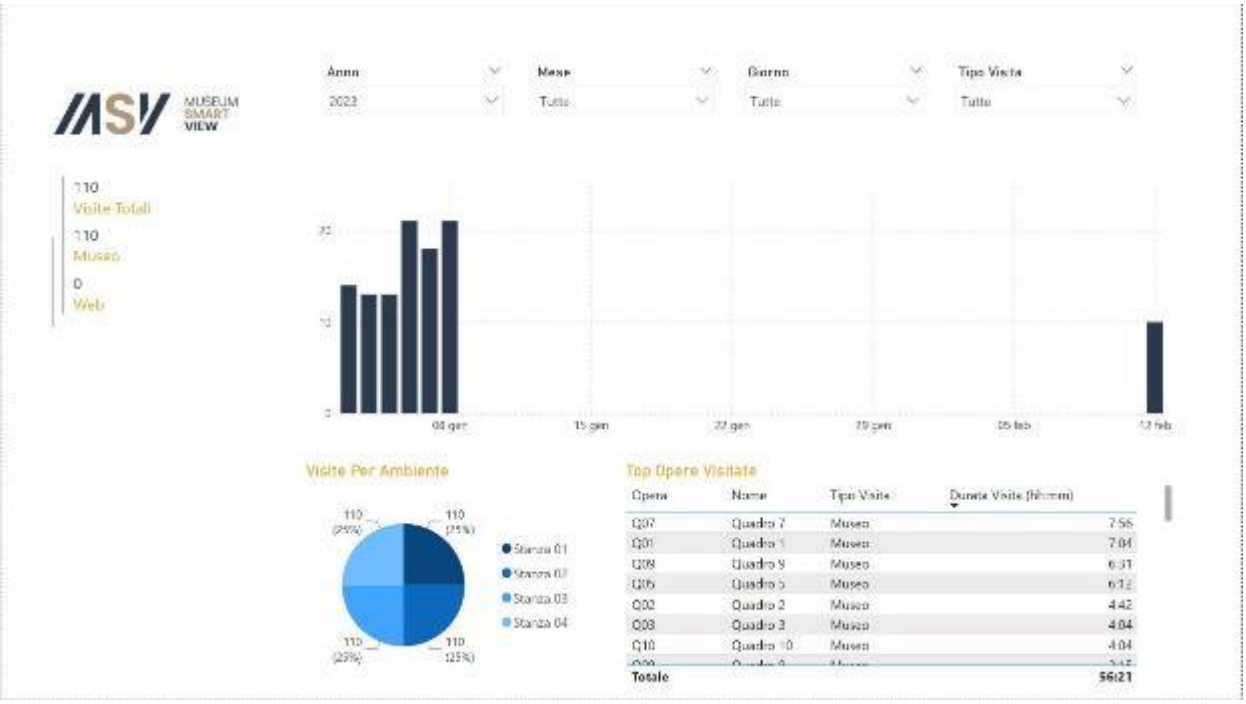


Figura 11: Dashboard Visite/ambienti/opera

## 5.2. Dashboard profilo Visitatori/Utenti

### 5.2.1. Obiettivo

La dashboard è stata progettata per dare all'utente amministratore del museo uno spaccato degli utenti/visitatori e delle principali caratteristiche del loro profilo

### 5.2.2. Descrizione

La dashboard permette di gestire il filtro sull'anno e sulla tipologia di visitatore (web o presso museo).

A seconda del livello di profilazione dell'utente collegato (amministratore di un singolo dipartimento, amministratore di un intero museo, super amministratore di un'aggregazione di musei) saranno caricati i dati relativi.

Sostanzialmente vengono visualizzati:

- Il totale degli utenti/visitatori ed i relativi subtotali per tipologia
- Il dettaglio delle nazioni di provenienza
- Lo spaccato per fasce di età
- Il dettaglio per genere
- Il dettaglio per titolo di studio

L'utente attraverso l'opzione di drill insita in PowerBI, cliccando su qualsiasi entità grafica della dashboard potrà aggiungere un ulteriore filtro rigenerando al volo la visualizzazione della stessa (es: cliccando su una "fetta" rappresentativa di una nazione della torta delle Nazioni, la dashboard si ricaricherà con i dati relativi ai visitatori di quella Nazione).

Figura 12: Dashboard profilo Visitatori/Utenti

## 5.3. Dashboard catalogo Opere

### 5.3.1. Obiettivo

La dashboard è stata progettata per dare all'utente amministratore del museo un quadro completo delle opere catalogate

### 5.3.2. Descrizione

La dashboard permette di gestire il filtro sull'anno e sulla tipologia di opera.

A seconda del livello di profilazione dell'utente collegato (amministratore di un singolo dipartimento, amministratore di un intero museo, super amministratore di un'aggregazione di musei) saranno caricati i dati relativi.

Sostanzialmente vengono visualizzati:

- Il totale delle opere
- Il dettaglio delle opere per autore
- Il dettaglio delle opere per datazione
- Il dettaglio delle opere per provenienza
- Le prime 10 opere più visitate

L'utente attraverso l'opzione di drill insita in PowerBI, cliccando su qualsiasi entità grafica della dashboard potrà aggiungere un ulteriore filtro rigenerando al volo la visualizzazione della stessa (es: cliccando su una "fetta" rappresentativa di un autore della torta degli autori, la dashboard si ricaricherà con i dati relativi alle opere di quell'autore).

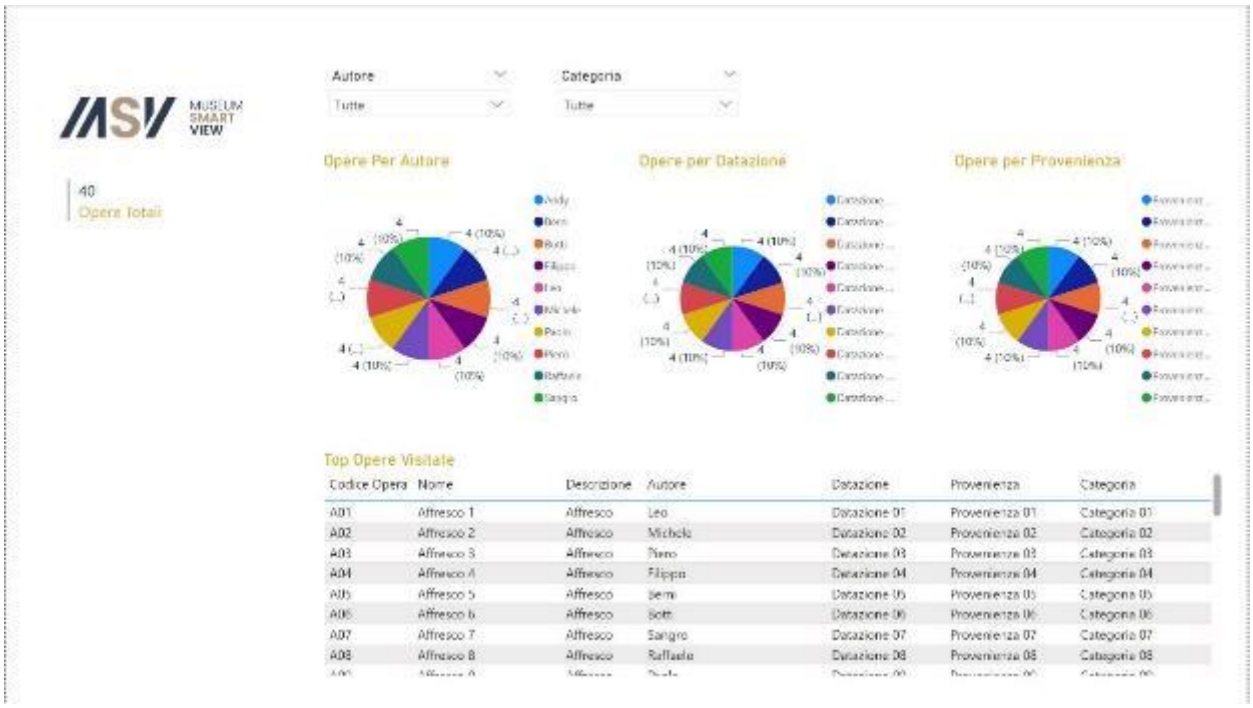


Figura 13: Dashboard catalogo Opere

## 6. Scenari futuri (AI e servizi cognitivi)

Nell'era digitale, il volume dei dati degli utenti generati nei musei, aumenta in modo esponenziale. Oggi, un'istituzione culturale dispone di molteplici punti di contatto digitali dai quali estrarre dati sui suoi visitatori: sul suo sito web, tramite le pagine sui social media, attraverso il ticketing, le applicazioni mobili e la sensoristica da campo.

L'applicazione dell'AI per analizzare i dati su visitatori e visite, offrirà ai musei nel futuro prossimo l'opportunità di conoscere e studiare meglio il proprio pubblico al fine di creare e proporre esperienze ancor di più personalizzate e coinvolgenti.

Gli algoritmi di AI offrono già la possibilità di esaminare i dati esistenti, oltre a facilitare la raccolta di nuovi.

La sfida sarà utilizzare l'apprendimento automatico come fattore abilitante per l'elaborazione di nuovi modelli di visita e prevedere le tendenze future, in un campo che necessita di mantenersi al passo con i tempi ed attrarre le nuove generazioni già abituate a paradigmi diversi di fruizione dei contenuti.

## 7. Riferimenti Bibliografici


| TITOLO DOCUMENTO | VER. | ANNO |
|---|---|---|
| *Intelligenza artificiale e patrimonio culturale: algoritmi che cambiano scenari e prospettive* di Camilla Donà delle Rose | 1.0 | 2021 |

fine del documento