



**UNIVERSITA' POLITECNICA DELLE MARCHE**

**FACOLTA' DI INGEGNERIA**

---

Corso di Laurea triennale in Ingegneria Gestionale

**Digital twin e strumenti CAD: rappresentazione e simulazione di reti IoT in  
diversi ambiti di applicazione**

**Digital twin and CAD tools: representation and simulation of IoT networks  
in different fields of application**

Relatore: Chiar.ma

Prof.ssa *Maura Mengoni*

Correlatore: Chiar.mo

Dott. Alex Altieri

Tesi di Laurea di:

*Marco Pierandrei*

A.A. 2019 / 2020

# **Indice**

## **Indice delle immagini**

## **Introduzione**

### **1. Digital Twins**

#### **1.1 Storia del Digital Twin**

#### **1.2 Definizioni di Gemello Digitale**

#### **1.3 Tecnologie Abilitanti**

#### **1.4 Campi di applicazione di un Gemello Digitale**

##### **1.4.1 Progettazione**

##### **1.4.2 Produzione**

##### **1.4.3 Post-Vendita**

#### **1.5 Sfide da affrontare**

##### **1.5.1 Connettività**

##### **1.5.2 Hardware**

##### **1.5.3 Sicurezza**

##### **1.5.4 Risorse umane**

### **2 Microsoft Azure Digital Twin**

#### **2.1 Introduzione**

#### **2.2 Caso studio**

#### **2.3 Microsoft Azure**

#### **2.4 Gemelli digitali con Azure**

##### **2.4.1 Modelli**

#### **2.5 Configurazione di un'istanza**

#### **2.6 Creazione di una Applicazione Client**

**2.6.1 Autenticazione per il servizio**

**2.6.2 Caricare un modello**

**2.6.3 Creazione dei Gemelli Digitali**

**2.6.4 Creazione delle relazioni**

**2.6.5 Query sui Gemelli Digitali**

**2.7 Approfondire i Gemelli Digitali con un'app client**

**2.7.1 Modellare un ambiente fisico**

**2.7.2 Avvio dell'app**

**2.7.3 Caricamento dei modelli**

**2.7.4 Errori**

**2.7.5 Creazione dei Gemelli**

**2.7.6 Creazione del grafo e specifica relazioni**

**2.8 Soluzione end-to-end**

**2.8.1 Creazione dell'istanza**

**2.8.2 Configurare l'app**

**2.8.3 Elaborazione dei dati simulati da un dispositivo Hub IoT**

**2.8.4 Propagazione eventi di Gemelli Digitali tramite il grafo**

### **3. Sviluppi futuri**

**3.1 Introduzione**

**3.2 Evoluzione del progetto**

**3.3 Modelli 3D**

**3.4 Sensoristica**

**3.4.1 Face Recognition e telecamere**

**3.4.1.1 Trovare i volti nell'immagine**

**3.4.1.2 Analisi delle caratteristiche facciali**

**3.4.1.3 Codifica dei volti**

**3.4.1.4 Identificare la persona raffigurata nell'immagine**

**3.4.1.5 Avvertenze**

**3.4.2 Human Activity Recognition (HAR)**

**Conclusione**

**Ringraziamenti**

## Introduzione

In questa tesi verrà illustrato in linea generale quello che è un Digital Twin dalla sua nascita fino ad oggi, poi si entrerà un po' più nel dettaglio andando a dare delle prime definizioni di gemelli digitali, poi si procederà con una analisi di come si realizza in gemello digitale, quali sono le tecnologie abilitanti e dove questo è possibile applicarlo, in quest'ultima parte c'è da dire che nel tempo i campi di applicazione sono aumentati e in maniera esponenziale.

Il Digital Twin sta rivoluzionando il mondo delle imprese ma anche la vita quotidiana delle persone. Infatti molto spesso si sente parlare di gemelli digitali applicati impiegati nelle aziende, e molto poco per quanto riguarda la loro applicazione negli ambienti domestici ma anche città e molto altro.

Per questo è stato scelto di provare a gettare quelle che sono le basi per la realizzazione di un Digital Twin di una stanza di una abitazione, il quale poi con tutte le evoluzioni future applicabili al mio progetto può diventare un vero e proprio Gemello Digitale che rende la casa più tecnologica e migliora la vivibilità dell'ambiente.

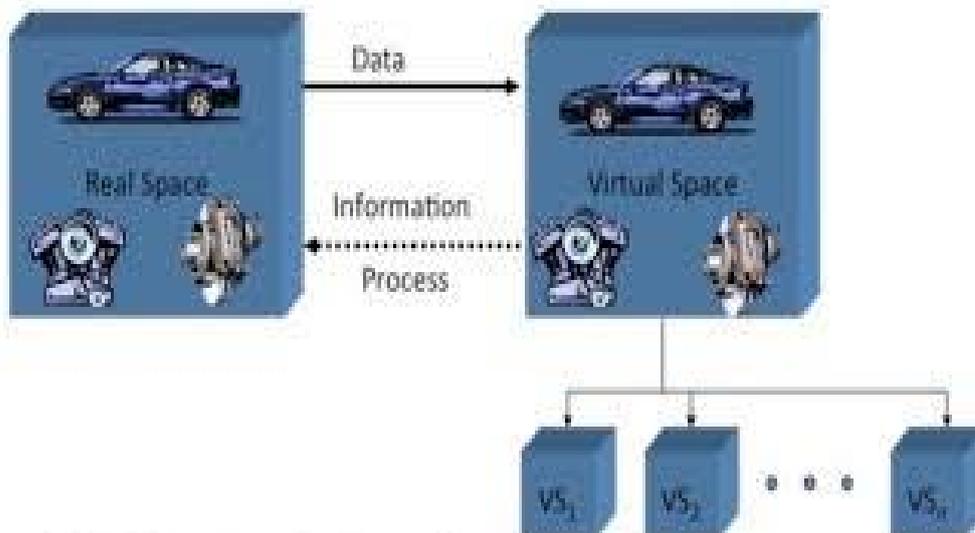
Per questa realizzazione è stato necessario l'impiego di una piattaforma tramite la quale sviluppare il progetto, per questo è stata scelta Microsoft Azure, tramite la quale è stato possibile realizzare tutto il nostro progetto. Dopo aver effettuato l'accesso alla piattaforma si è proceduto con la creazione delle risorse, di sensori, dei modelli ecc. poi una volta collegato tutto insieme è stata effettuata la simulazione finale e completa per verificare la perfetta esecuzione del sistema.

Per quanto riguarda il futuro l'idea è quella di applicare un maggior numero di sensori dai meno sofisticati ai più complessi ma che ci permettono di acquisire dati di fondamentale importanza per centrare l'obiettivo finale, cioè quello di andare a fare delle analisi sul comportamento delle persone presenti nella stanza. Giocano un ruolo importantissimo in questa partita anche gli algoritmi che verranno sviluppati per riuscire a fare questo tipo di analisi. Riuscire in un progetto simile significa migliorare la qualità della vita delle persone, aumentare l'efficienza delle abitazioni, la loro sicurezza.

# 1. Digital Twins

## 1.1 Storia del Digital Twin

Il concetto di Digital Twin è stato originariamente teorizzato nel 2002, all'Università del Michigan, dal Dott. Micheal Grieves, nel contesto del Product Lifecycle Management (PLM). Il Dott. Grievens lo chiamò "Conceptual Ideal for PLM" e lo definì come "a virtual, digital equivalent to a physical product". Esso si basa sull'idea che un sistema fisico possa essere ricreato in digitale attraverso le informazioni e le logiche che lo caratterizzano. I due sistemi, fisico e digitale, rimangono strettamente accoppiati durante l'intero ciclo di vita ed evolvono insieme, è proprio per questo motivo che colui che teorizzò il tutto inserì all'interno del nome l'acronimo PLM. Le informazioni elaborate dalla copia digitale si configurano come "gemelle" delle informazioni del sistema fisico. Il vantaggio risiede nel fatto che, in condizioni ideali, qualsiasi informazione che si possa ottenere dall'analisi di un bene o da un processo reale possa essere ottenuta attraverso l'interrogazione del suo gemello digitale. Pertanto, il Digital Twin è definito come un insieme di informazioni virtuali che descrivono completamente un prodotto, un sistema o un processo fisico, permettendo di valutarne lo stato o il funzionamento attuale e futuro. Il modello nel tempo cambiò nome ma non concetto e nel 2006, venne denominato MODELLO DI MIRRORING DELLE INFORMAZIONI.[1]



Dr. Michael Grieves, University of Michigan, Lone Engineering Center, Dec. 3, 2001

Fig.1

Il termine Digital Twin è stato adottato nel campo aeronautico ed aerospaziale nel 2010 dalla NASA che ne ha dato origine. Gli scienziati della NASA già nel 1970 crearono inconsapevolmente perché non hanno affatto utilizzato l'Internet of Things quello che oggi noi chiamiamo Digital Twin, a causa del mancato sbarco sulla luna, in seguito ad un incendio che provocò gravi danni agli equipaggiamenti, si ritrovarono a dover risolvere problemi a vari migliaia km di distanza dalla terra. Venne creato dunque un Digital Twin, gemello della navicella, utilizzato per cercare di riprodurre diversi scenari sulla terra con lo scopo di supportare gli astronauti nella realizzazione di decisioni in situazioni critiche. La definizione data dagli stessi scienziati al Digital Twin fu quanto segue:

“Una simulazione ultra-realistica ad alta scalabilità, che utilizza i migliori modelli fisici disponibili, i dati dei sensori e quelli storici per il mirroring di uno o più sistemi reali” (NASA, 2010). Verranno poi realizzati dei gemelli digitali da parte di AFRL ( Air Force Research Laboratory) anche per aerei militari di ultima generazione.

A livello aziendale il primo Digital Twin fu applicato nell'Industria 4.0 dalla Siemens, da qui in avanti moltissimi studiosi iniziarono a fare ricerche e pubblicazioni che in poco sono cresciute di numero in maniera esponenziale.

Il Digital Twin rappresenta un passaggio importante nella storia della tecnologia, grazie allo studio sui gemelli digitali si è riusciti a superare molti limiti come ad esempio l'acquisizione dei dati, processare enormi quantità di dati, performance limitate da parte degli elaboratori e sviluppo di nuovi algoritmi più efficienti .



**Fig.2**

## 1.2 Definizioni di Gemello Digitale

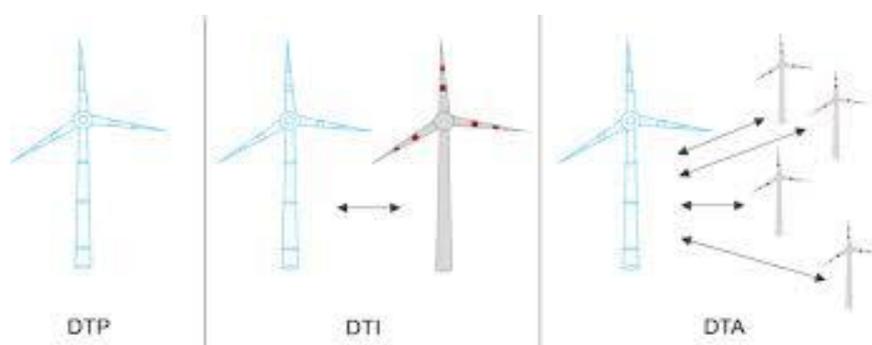
**Digital Twin (DT):** E' la rappresentazione virtuale di un'entità fisica, vivente o non vivente, di una persona o di un sistema anche complesso. Il sistema digitale è in qualche modo connessa con la parte fisica, con la quale può scambiare dati e informazioni, sia in modalità sincrona (in tempo reale), che asincrona (in tempi successivi). Poiché il concetto di Digital Twin abbraccia l'intero ciclo di vita dei prodotti che consistono in creazione, costruzione, utilizzo / supporto e smaltimento, esistono diversi tipi di Digital Twin.

**Digital Twin Prototype (DTP):** Comprende progetti, analisi e processi orientati alla realizzazione di un bene fisico, gemello di quello virtuale. Quindi il DTP contiene tutte quelle che sono le informazioni necessarie per descrivere e produrre un modello fisico, il quale non è collegato ad una particolare risorsa materiale.

**Digital Twin Instance (DTI):** A differenza del DTP il DTI è collegato ad una particolare risorsa materiale ben specificata infatti esso rappresenta il gemello di ogni specifico prodotto realizzato, con un collegamento che ha durata pari a quella del ciclo vita del prodotto.

**Digital Twin Environment (DTE):** Uno spazio applicativo di fisica multi dominio integrato per operare su Digital Twins. Ovvero interrogazione di DTI per stati attuali e passati e previsione del comportamento futuro e delle prestazioni dei prodotti fisici in vari scenari utilizzando il loro DTI (e possibilmente DTA).

**Digital Twin Aggregate (DTA):** E' l'aggregazione dei DTI, i cui dati e informazioni, digitali, possono essere utilizzati per interrogare il prodotto fisico.



**Fig.3**

### 1.3 Tecnologie Abilitanti

Le tecnologie abilitanti o anche dette KET “Key Enabling Technologies”, sono un ponte per la crescita e l’occupazione e la Commissione Europea le ha definite come “tecnologie ad alta intensità di conoscenza e associate a elevata intensità di ricerca e sviluppo (R&D), a cicli di innovazione rapidi, a consistenti spese di investimento e a posti di lavoro altamente qualificati. Rendono possibile l’innovazione nei processi, nei beni e nei servizi in tutti i settori economici e hanno rilevanza sistematica”.

Il Digital Twin è esso stesso una tecnologia abilitante ma allo stesso tempo necessita per il suo funzionamento di altre KET, che possono essere raggruppate in tre macro gruppi: CONNETTIVITA’, DIGITALIZZAZIONE, INTELLIGENZA ARTIFICIALE (AI).

Scomponendo questi gruppi ed andando nel dettaglio, cercheremo di capire quali sono quelle tecnologie e strumenti che consentono la realizzazione di un Digital Twin.

**SENSORI:** Sono degli strumenti fondamentali, hanno lo scopo di catturare quelle che sono le informazioni e i dati operativi e ambientali che poi verranno analizzati.

**BIG DATA:** Per definizione questi dati che vengono raccolti Tutti questi dati che vengono raccolti dai sensori, vengono raggruppati con anche altri dati aziendali, fortunatamente oggi i costi di memorizzazione dei dati si sono notevolmente abbassati, quindi anche sono accessibili anche alle PMI.

**INTERNET of THINGS:** La comunicazione tra il mondo fisico, cioè i dati che i vari sensori rilevano, ed il mondo digitale ma anche il passaggio contrario, cioè da virtuale a reale, avviene grazie ad una tecnologia di integrazione.

**SIMULATION MODELING:** La modellazione si riferisce alla trasformazione di oggetti e spazi fisici in modelli digitali che possono essere elaborati, studiati e gestiti da computers. Il modello digitale lavora in tempo reale con il sistema reale. Le tecnologie utilizzate per la realizzazione di questi modelli sono differenti a seconda che la riproduzione sia ad esempio di un macchinario o di una linea di produzione.

**INTELLIGENZA ARTIFICIALE:** Tramite l’AI le macchine riescono ad apprendere dalle esperienze passate, si adattano a dei nuovi input e a compiere delle attività come quelle umane, questo tipo di tecnologia oggi è possibile sfruttarla grazie alla capacità che abbiamo di radunare

ed elaborare enormi quantità di dati. L'Intelligenza Artificiale, riesce ad analizzare la situazione e prendere decisioni in maniera da ottimizzare il processo e raggiungere l'obiettivo.

**DATA ANALYTICS:** Sono delle tecniche che vengono utilizzate per analizzare tramite degli algoritmi, tutti quei dati che vengono rilevati dai sensori. Grazie al loro impiego è possibile comprendere l'evoluzione del sistema e consente quindi di prendere delle decisioni in maniera più sicura e precisa riducendo il fattore di errore.

Se vogliamo dare qualche numero in percentuale, stimati in base alle varie letture ed approfondimenti, delle tecnologie più utilizzate possiamo dire che l'IoT è utilizzata nel 67% poi seguono i Big Data & Data Analytics con il 59%, Simulation Modeling 50%, Artificial Intelligence 25%.

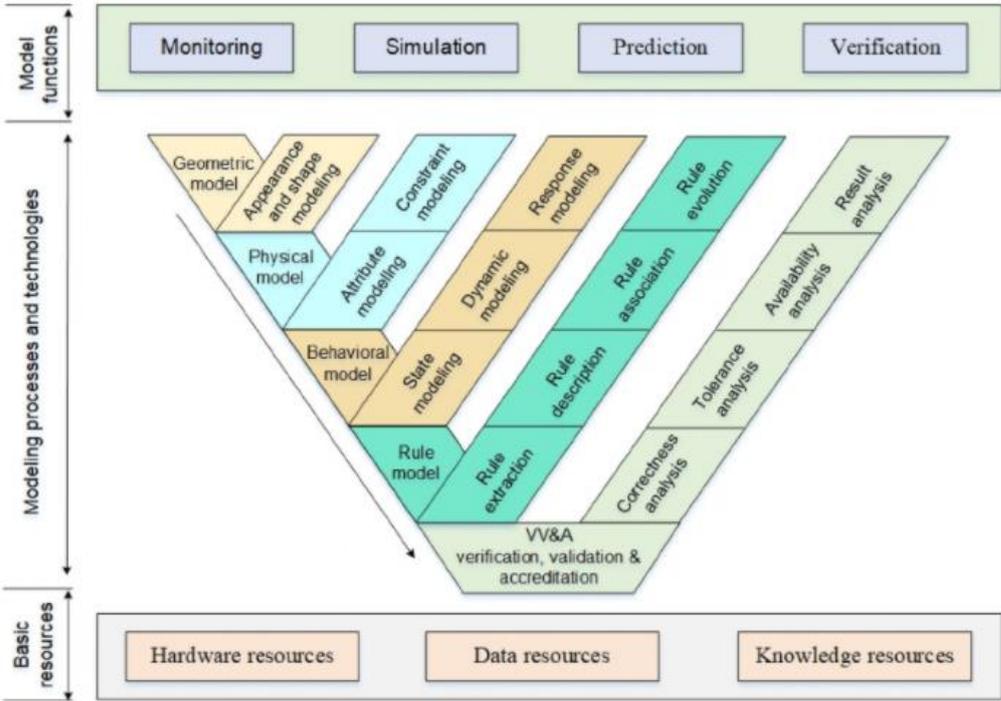


Fig.4

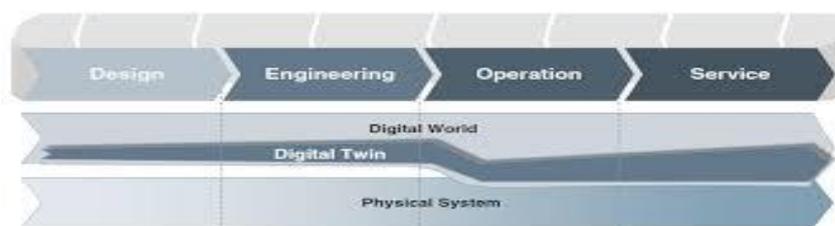
## 1.4 Campi di applicazione di un Gemello Digitale

Come abbiamo visto nei capitoli precedenti, quelli relativi alla storia del Digital Twin, tutto nasce durante un corso universitario di Product Lifecycle Management (PLM). Il nostro scopo ora è quello di capire l'impatto che il Digital Twin provoca nelle varie fasi del Ciclo Vita del prodotto. Digitalizzare il Ciclo Vita, significa avere la possibilità di lavorare in un ambiente virtuale nel quale non si utilizzano più né prototipi fisici né si faranno più prove fisiche sul prodotto ma bensì si lavorerà con la sua identità virtuale, molto più maneggevole e versatile.

Alcuni dei principali vantaggi che il Digital Twin offre sono:

- **Costo**, è nettamente più economico costruire un modello digitale piuttosto che uno fisico sul quale effettuare delle simulazioni.
- **Tempo**, la velocità di un modello di simulazione dipende dalle sue dimensioni, dalla complessità e dal tipo di tecnologia che viene utilizzata ma sicuramente sarà sempre più veloce delle simulazioni su un modello fisico.
- **Assistenza**, riuscire ad essere sempre vicini al cliente migliorando il servizio offerto.
- **Formazione** dei dipendenti molto più efficiente, hanno la possibilità di simulare e provare qualsiasi tipo di modifica in ogni fase del ciclo vita, in un ambiente virtuale con dei dati che possono essere anche quelli raccolti nell'ambiente reale, aumentando quindi la qualità della loro formazione.
- **Ripetibilità**, tutte le simulazioni e gli esperimenti possono essere ripetuti sempre alle stesse condizioni grazie al mondo virtuale, mentre se si deve fare simulazioni su un modello fisico, le condizioni in cui si effettueranno i vari test non saranno mai le stesse se ripetuti.

Ancora oggi però si fa fatica ad avere una riproduzione perfetta in digitale di quello che è l'oggetto fisico reale, perché riuscire ad abbinare tutte le caratteristiche dell'ambiente in termini di fisica, termica, elettronica e chimica è una sfida molto costosa e difficile. Per questo motivo se si intende adottare questo tipo di tecnologia, è necessario valutare in quale fase del ciclo vita inserirla (Progettazione, Produzione, Post-vendita).



**Fig.5**

### 1.4.1 Progettazione

Qualora il Digital Twin, venisse adottato in fase di progettazione è possibile utilizzarlo in diverse maniere come la prototipazione di nuove idee, utilizzando tutte quelle che sono test virtuali, che possono servirsi anche di dati reali precedentemente rilevati, i quali rendono la simulazione molto realistica e veritiera, ed aiutano i progettisti a rilevare eventuali errori ed anomalie del prototipo. Grazie all'IoT c'è la possibilità di mantenere uno stretto collegamento tra il progettista ed il prodotto grazie ai sensori che sono collegati all'ambiente in esame, quest'ultimi forniscono in maniera costante e continua dati ed informazioni. Per quanto riguarda il risparmio basta pensare ad una azienda automobilistica che per testare l'aerodinamicità dei propri modelli necessita di una galleria del vento, utilizzando prototipi virtuali e simulazioni virtuali con dati reali, risparmiano in termini di tempo e soldi.

### 1.4.2 Produzione

Per quanto riguarda la produzione lo sviluppo di Digital Twin in questa fase del ciclo vita, può portare di vantaggi per quanto riguarda la collaborazione dei vari team di lavoro nel reparto produttivo; per gestire ed ottimizzare al meglio quelle che sono le linee produttive confrontate rispetto al layout, a quelli che sono i flussi materiali e informativi e ai processi tutto svolto in tempo reale, eliminando quelle che sono le inefficienze nelle fasi di lavorazione. In questa fase l'applicazione di queste tecnologie tendono ad incrementare la produttività e quindi anche la qualità dei prodotti.

### 1.4.3 Post-Vendita

Una volta che il prodotto raggiunge il cliente, questa tecnologia può essere utilizzata per la configurazione e regolazione del bene da remoto, riducendo quelli che sono i costi di installazione e di aggiornamento, inoltre può essere utilizzata per la raccolta dati relativi a quelle che sono le prestazioni del macchinario per effettuare delle previsioni, individuazione di eventuali problemi, in maniera tale che vengono eseguiti gli interventi di manutenzione mirata e con anticipo rispetto al guasto, assicurando così al cliente da eventuali danni maggiori. Qualora si sia verificato un guasto grazie al Digital Twin è possibile capire quelli che sono gli elementi guasti, quindi l'operatore che si occuperà della sostituzione avrà già a conoscenza il guasto e minimizzerà il tempo di fermo macchina.

L'applicazione del Digital Twin non è un'esclusiva del mondo dell'industria e della produzione di beni ma vengono utilizzati anche nel campo dei servizi come il Trasporto, per avere dei gemelli che come per l'industria ci aiutano a studiare i miglioramenti da apportare, fornendoci continuamente dati e consentono delle analisi predittive; nel mondo campo del Retail, per avere dei gemelli dei clienti e studiare i loro comportamenti nelle differenti campagne marketing, inoltre servono per capire come pianificare al meglio le attività in-store; nella Sanità grazie a dei dati generati dell'IoT, possono svolgere dei ruoli fondamentali in termini di: risparmio sulla spesa sanitaria, assistenza sanitaria personalizzata per ogni paziente; Smart Home, che non si tratta del semplice accendere e spegnere le luci, ma è qualcosa di molto più sofisticato che è caratterizzata dalla connessione in rete e dalla presenza di nuove caratteristiche, come ad esempio i comandi vocali, telecamere che riconoscono i volti ed i movimenti delle persone. Questo tipo di tecnologia viene adottata per migliorare la qualità della vita, aumentare l'efficienza nelle abitazioni consentendo anche di risparmiare nei consumi.



**Fig.6**

## 1.5 Sfide da affrontare

In questo paragrafo si cerca di riassumere tramite un elenco, quelle sfide che coloro i quali lavorano in questo ambito dagli anni '60 ad oggi sono chiamati ad affrontare.

### 1.5.1 Connettività

Questo concetto viene considerato una sfida perché, riuscire a mantenere sempre connessi in tempo reale oggetti o persone, in movimento o statici non è sempre scontato e semplice, specialmente se essi si trovano in zone del globo remote o poco sviluppate a livello tecnologico. Inoltre altra grande sfida è quella legata all'abbattimento dei costi, per avere dei dati reali senza errori è necessario essere dotati di una larghezza di banda adeguata ed una potenza di calcolo elevata i cui costi potrebbero essere non indifferente.

### 1.5.2 Hardware

Come è stato già detto i sensori sono degli elementi essenziali per la buona riuscita del gemello digitale, più ne vengono inseriti, logicamente con criterio senza cadere nella trappola dello spreco, più dati noi riusciamo a catturare, le problematiche sorgono quando si devono collegare un numero significativo di sensori, quindi la loro gestione e relativa manutenzione.

### 1.5.3 Sicurezza

I dati sono la parte più sensibile e vulnerabile all'interno del Digital Twin, costituiscono un grandissimo valore per l'impresa, quindi è necessario adottare sistemi di salvataggio sicuri per non smarrirli o danneggiarli ed allo stesso tempo per proteggerli da eventuali manomissioni o furti.

### 1.5.4 Risorse umane

Il personale addetto a lavorare con questa tecnologia, deve essere altamente qualificato con una buona formazione in materia alle spalle, a sua volta questo tipo di investimenti in ambienti in cui non sono presenti delle figure prettamente tecniche ed addestrate in materia, scaturisce una crescita a livello tecnico dei dipendenti i quali faranno parte della trasformazione dell'azienda.

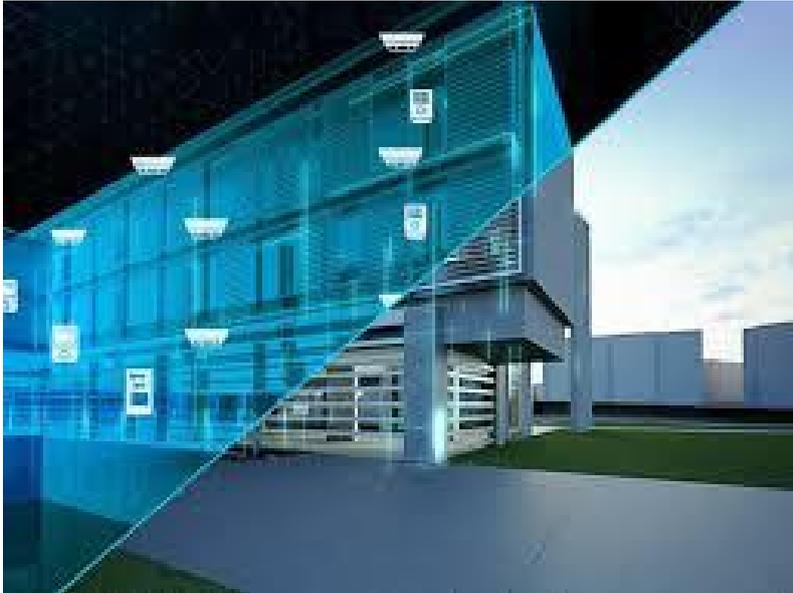
## **2. Microsoft Azure Digital Twin**

### **2.1 Introduzione**

In questo capitolo verrà descritto, quello che è stato appreso e realizzato in circa due mesi di lavoro, utilizzando la piattaforma Azure della Microsoft per la creazione e la simulazione di un Digital Twin. Tutto ciò che verrà rappresentato nelle pagine a seguire è il frutto di studio e lavoro in di una disciplina in fase di evoluzione, il mio compito è stato quello di imparare le basi della materia ed iniziare a gettare quelle che potranno essere le fondamenta di un progetto più ampio e complesso.

### **2.2 Caso studio**

Il mio caso studio come detto sopra è stato quello di trattare le basi dell'implementazione di un Digital Twin in un ambiente fisico, quale esso sia una casa, un ufficio, uno studio ecc... Prima di iniziare a fare qualsiasi passo è stato necessario studiare nei particolari quello che effettivamente rappresenta un Digital Twin ed anche le tecnologie con cui è possibile sviluppare un sistema del genere. Una volta capito il funzionamento dello strumento ed il campo di applicazione, si è iniziato a pensare a come impostare il problema, dalla scelta della piattaforma con la quale sviluppare il progetto allo schema dettagliato dello stesso. Il primo passo è stato la scelta di Microsoft Azure, perché se confrontata con la sua concorrente AWS (Amazon Web Services) si noterebbero differenze sostanziali quali assistenza, Microsoft dispone di una quantità di documenti ed esempi ben maggiori rispetto Amazon, semplicità dell'utilizzo dell'interfaccia nel portale, inoltre Microsoft mette a disposizione una sottoscrizione gratuita per gli studenti, tramite la quale è possibili utilizzare alcune specifiche risorse tra cui anche il Digital Twin. Fatto il primo passo della scelta si è iniziato a pensare a quale fosse l'obiettivo finale, cioè la creazione di un ambiente virtuale capace di acquisire informazioni dall'ambiente reale ed elaborarle. L'acquisizione delle informazioni avviene tramite l'utilizzo di sensori, in questo caso si è simulato lo studio di due parametri quali temperatura ed umidità presenti all'interno di una stanza, questa stanza fa parte di un piano di un edificio. Una volta che le informazioni reali vengono trasformate in digitale dal sensore, vengono in maniera continua, trasferite nel mondo virtuale cioè al gemello della stanza, in cui tramite delle apposite strutture, preventivamente realizzate, questi dati vengo elaborati per ottenere il risultato finale, che il nostro gemello inizi ad elaborare informazioni.



**Fig.7**

### 2.3 Microsoft Azure

Azure è la piattaforma pubblica di cloud computing della Microsoft ciò consente la distribuzione di servizi di calcolo come server, database, software, analisi e intelligence, il tutto tramite il cloud, in maniera da offrire uno sviluppo rapido, risorse flessibili ed economie di scala. Quindi non si intende più il termine vago che viene spesso applicato ai servizi di consumo che memorizzano i dati su un server remoto in qualche posto lontano, ma si considera questo servizio atto ad aziende, organizzazioni ed addirittura individui che intendono lavorarci.

Questa tecnologia rappresenta una vera rivoluzione a livello industriale nella visione delle risorse IT. Essa offre anche numerosi vantaggi quali per prima cosa i COSTI, acquistando licenze Azure, libera colui che investe dalla spesa per parti hardware, software, configurazione e tutto ciò che concerne il mantenimento delle stesse. VELOCITA', tutta la parte dei servizi cloud computing viene offerta in maniera tale che si possano utilizzare anche delle grandi quantità di risorse di calcolo, richiedendole con un semplice click del mouse. PRODUTTIVITA', il team IT aziendale non sarà più occupato a svolgere dei compiti che impiegano parecchio tempo e risorse, ma riesce a focalizzarsi meglio per il raggiungimento degli obiettivi aziendali. AFFIDABILITA', il cloud computing permette di ridurre i costi di backup delle informazioni e ne aumenta la semplicità. SICUREZZA, offre un elevato livello di sicurezza e protezione dei dati e app da eventuali inconvenienti. Per capire meglio questa tecnologia che utilizza Azure del cloud computing, utilizziamo un esempio; GOOGLE DRIVE utilizza il cloud computing, in questa applicazione possiamo salvare immagini e documenti e condividerli con altri utenti.

I servizi messi a disposizione da Azure sono moltissimi e spaziano in diverse categorie dal Machine Learning, al Data Analytics, alla creazione o replica di macchine virtuali.

Il servizio di Microsoft Azure può essere suddiviso in tre “pacchetti” che si distinguono in base alle modalità di erogazione necessaria, IaaS (Infrastructure-as-a-Service), SaaS (Software-as-a-Service), PaaS (Platform-as-a-Service). Azure dispone di una rete globale, copre 54 regioni in tutto il pianeta, molto più grande di qualsiasi altra azienda che offre servizi Cloud. Una percentuale molto elevata delle migliori aziende del mondo circa il 90%, utilizza Azure, quindi la qualità del servizio offerto e la sua flessibilità sono cavalli di battaglia della Microsoft, capaci di attrarre notevoli clienti. Altro particolarità della piattaforma è quella di consentire la creazione di applicazioni intelligenti, usando servizi avanzati per i dati e l’intelligenza artificiale, riuscendo così ad aumentare la flessibilità ed abbattendo molto i costi per il cliente.



**Fig.8**

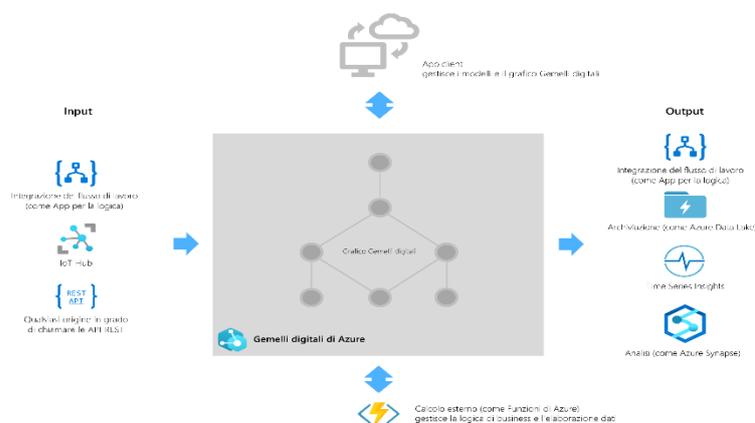
## 2.4 Gemelli Digitali con Azure

I Gemelli digitali di Azure sono un servizio PaaS che consente la creazione di grafici di conoscenze basati su modelli di ambienti interi, come fabbriche, case, fattorie, stadi e tantissime altre applicazioni, sfruttando la propria esperienza del settore oltre a quella di Azure per i Gemelli Digitali. Questa tecnologia ci dà la possibilità di:

- Modellare qualsiasi ambiente in maniera da poterci implementare gemelli digitali in modo sicuro e scalabile
- Connettere delle risorse come ad esempio i dispositivi IoT e i sistemi aziendali già esistenti
- Possibilità di utilizzare analisi dati e intelligenza artificiale per avere sempre a disposizione i dati del passato e quindi riuscire a fare previsioni future

La funzione Gemelli Digitali viene solitamente affiancata da altre applicazioni e servizi di Azure come parte di una soluzione IoT molto più allargata; noi nel nostro caso sperimentale abbiamo utilizzato:

- Una **istanza** del servizio Gemelli Digitali, tramite la quale ci era possibile archiviare il modello ed il grafo, con il relativo stato e gestire quindi l'elaborazione degli eventi
- Una **app client** che gestisce l'istanza, tramite la configurazione dei modelli e l'estrazione delle informazioni dal grafico dei gemelli
- Un **Hub IoT** per fornire praticità di gestione dei dispositivi e del flusso dati IoT
- Un' **App per la logica**, sarebbe un servizio downstream per gestire le attività e che ci consente di avere una archiviazione offline fidata, analisi



**Fig.9**

Tutte queste funzioni le andremo poi ad analizzare una ad una nel nostro caso studio.

## 2.4.1 Modelli

Tutto quello che è l'ambiente fisico, stanze, persone, oggetti, che deve essere reso digitale, con Azure verrà rappresentato tramite dei Modelli. Questi modelli possono essere pensati come un vocabolario per descrivere la propria funzione, ad esempio in un palazzo definiremo vari modelli come "edificio", "piano", "appartamento", "ascensore" ecc. e poi per ogni modello andremo a creare il gemello digitale specifico.

Essi vengono definiti con un linguaggio DTDL (Digital Twin Definition Language), è basato su JSON-LD ed è indipendente dal linguaggio di programmazione.

La definizione del modello avviene per livelli ed il primo di essi è l'**interfaccia**, esse incapsula al suo interno l'intero modello, allo stesso tempo l'interfaccia può contenere anche uno o vari dei seguenti elementi:

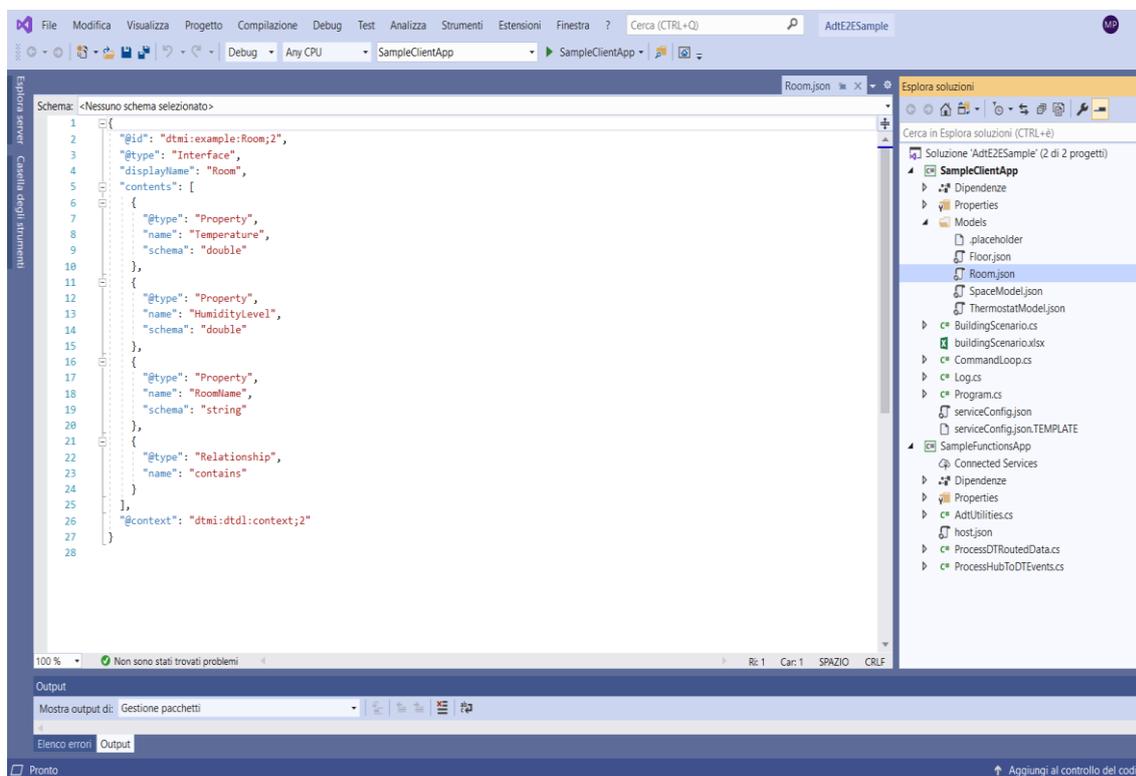
- **Proprietà** esse sono dei campi dati, le quali rappresentano lo stato di un'entità, provvedono all'archiviazione di backup e posso essere interrogate per la lettura in qualsiasi momento
- **Campi di telemetria**, questi rappresentano rilevazioni o eventi e vengono spesso utilizzati per descrivere le letture dei sensori del dispositivo. Se non si configura l'ascolto dell'evento e le azioni da intraprendere quando si verifica, non sarà presente alcuna traccia dell'evento qualora si decidesse di tornare indietro non sarà possibile rileggere quei dati. In questo caso a differenza delle proprietà, i dati non vengono archiviati nel suo gemello digitale, sono dati associati al tempo e devono essere gestiti non appena si verificano
- **Components**, essi consentono di compilare l'interfaccia del modello come assembly di altre interfacce, posso essere inoltre utilizzati per l'organizzazione ed il raggruppamento di un set di proprietà correlate all'interno di un'interfaccia del modello, in questo caso ogni componente è possibile considerarlo come una cartella dentro l'interfaccia
- **Relazioni** ci consentono di rappresentare il modo in cui un gemello digitale è o può essere associato ad altri gemelli, ci consentono di definire al meglio il grafico di tutte le varie entità correlate

Per essere garantita la compatibilità tra il modello DTDL e i Digital Twin di Azure è necessario che vengano rispettati determinati requisiti come ad esempio che tutti gli elementi di primo

livello di un modello devono essere di tipo INTERFACE; il Modello non deve definire alcun comando per il Gemello.

Quando un modello viene scritto è possibile farlo in qualsiasi editor di testo, come già detto il linguaggio DTDL segue una sintassi JSON, quindi è di fondamentale importanza quando si salva il modello, farlo con l'estensione JSON.

Nella seguente immagine verrà illustrato uno dei quattro modelli del nostro caso studio, il modello sarà quello della stanza chiamato "Room", nella parte destra dell'immagine è possibile notare quello che dicevamo poco fa, il modello è stato salvato con una estensione .JSON, mentre nella parte sinistra dell'immagine è possibile vedere tutta la sua struttura, dalle proprietà di Temperatura ed Umidità le quali ci devono restituire un valore numerico, alle relazioni che ci erano necessarie per il corretto funzionamento del gemello.



**Fig.10**

Mentre un pochino più lungo e complesso possiamo andare ad analizzare il modello del termostato, all'interno del quale troviamo delle proprietà e relazioni differenti da quelle presenti nel precedente modello, qui chiediamo al modello di dirci il nome, la posizione quindi se stanza quale e se piano anch'esso quale, poi viene chiesta la temperatura inoltre si richiede un indice di confort

Prima parte:

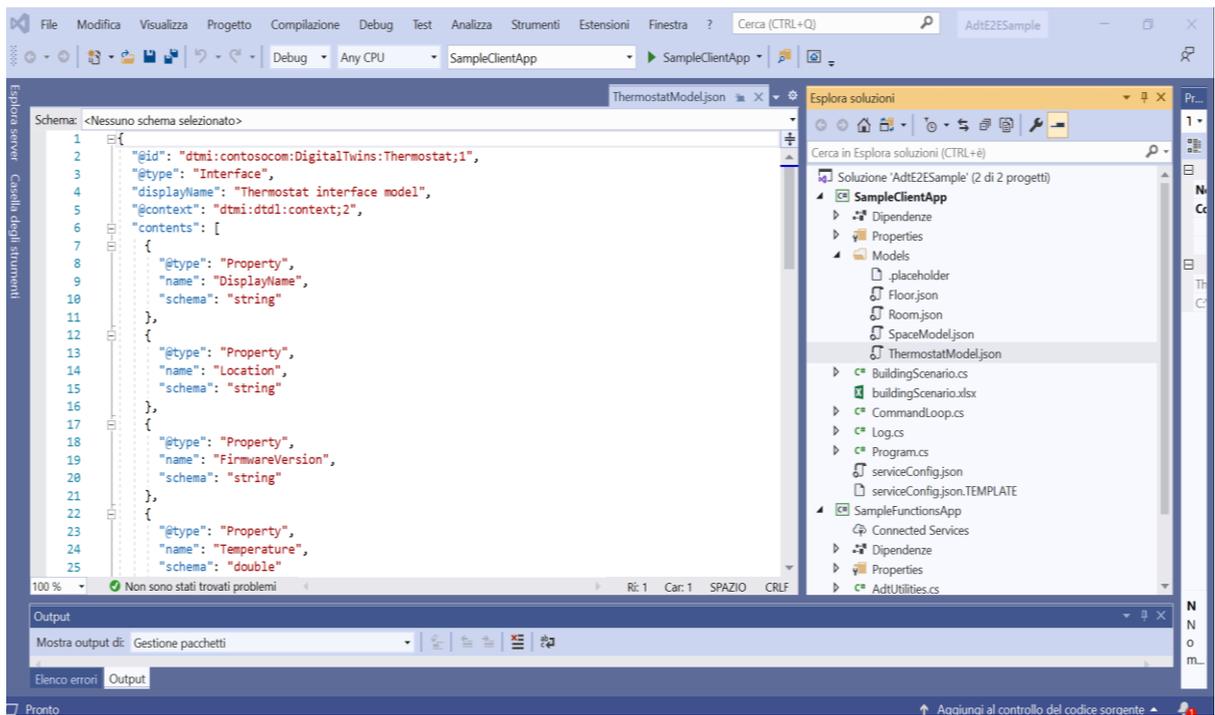


Fig.11

Seconda parte:

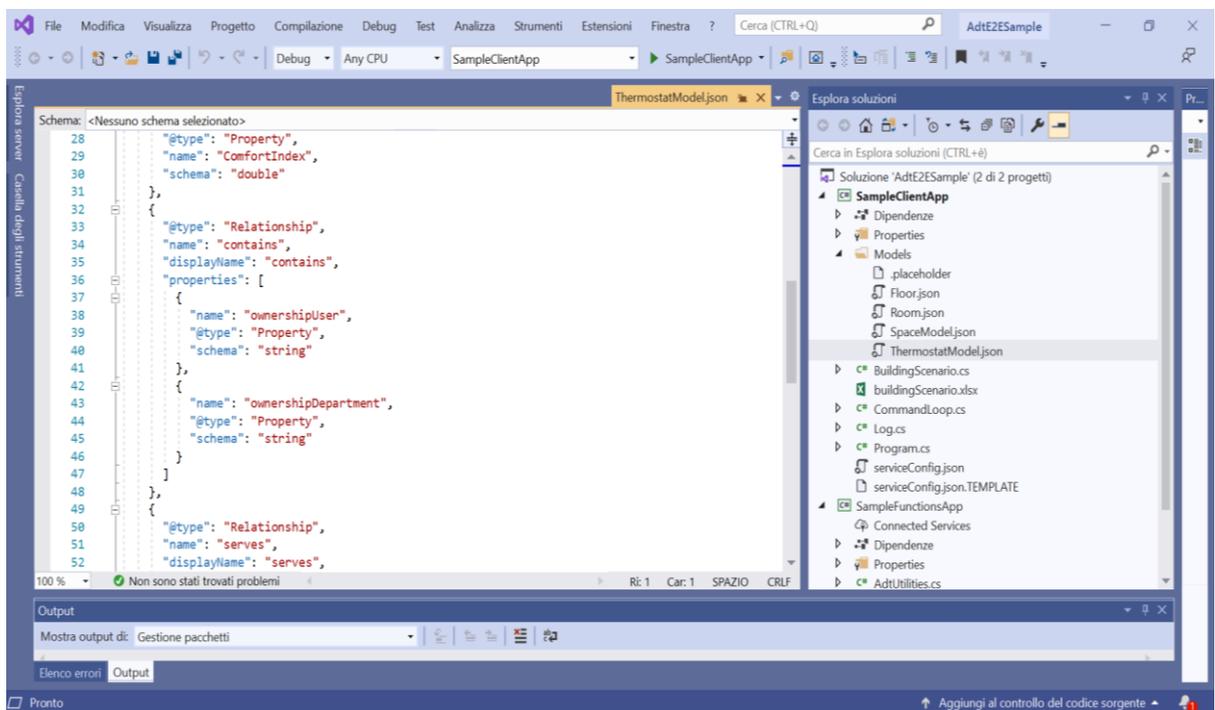


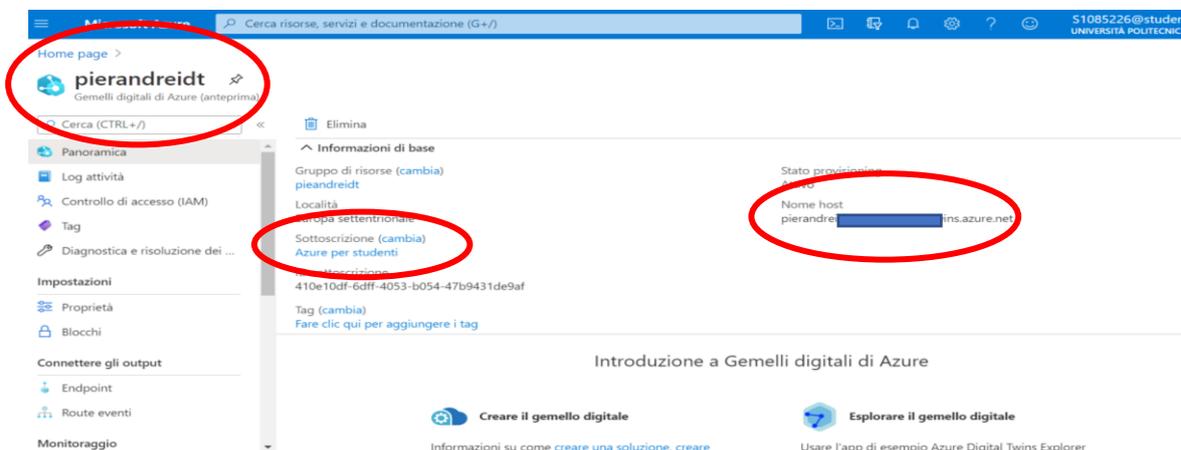
Fig.12

## 2.5 Configurazione di un'istanza

In questo paragrafo analizzeremo i vari passaggi che sono serviti per la configurazione dell'istanza per i Digital Twins. Per prima cosa ricordiamo cosa è un'istanza, è quell'elemento che ci permette di archiviare i modelli e il grafo dei nostri gemelli, inoltre gestisce ed elabora gli eventi.

La configurazione per una istanza Azure Digital Twins si suddivide in tre parti:

- **Creazione dell'istanza**
- **Organizzazione delle autorizzazioni di accesso da parte degli utenti;** coloro che intendono accedere e governare i dati devono possedere la qualifica di *proprietario (anteprima) di Azure Digital Twins* nell'istanza. I vari ruoli e assegnazioni vengono fatte dall'amministratore della relativa istanza. Nel nostro caso era presente un solo amministratore non che anche proprietario (anteprima).
- **Impostare le autorizzazioni di accesso alle applicazioni;** dato che avremo anche un app cliente (analizzata nello specifico nei prossimi paragrafi) che interagisce con l'istanza, dovremo autorizzare la stessa ad accedere ai dati dei nostri gemelli digitali, quindi una volta registrata anche l'app, prenderemo i suoi dati necessari per l'autorizzazione e li inseriremo nell'istanza.



**Fig.13**

Nell'immagine sopra è possibile notare all'interno dei cerchi rossi, partendo da destra verso sinistra abbiamo, il nome dell'istanza, il tipo di sottoscrizione a nostra disposizione e l'URL della nostra istanza, quest'ultimo è un dato da salvare perché verrà richiesto anche in altre circostanze.

Nell'immagini seguenti possiamo andare a verificare le autorizzazioni agli utenti e come dicevamo prima nel nostro il ruolo di proprietario (anteprima) e quello di amministratore viene ricoperto dalla stessa persona, cosa che ai fini pratici non provoca nessun errore.

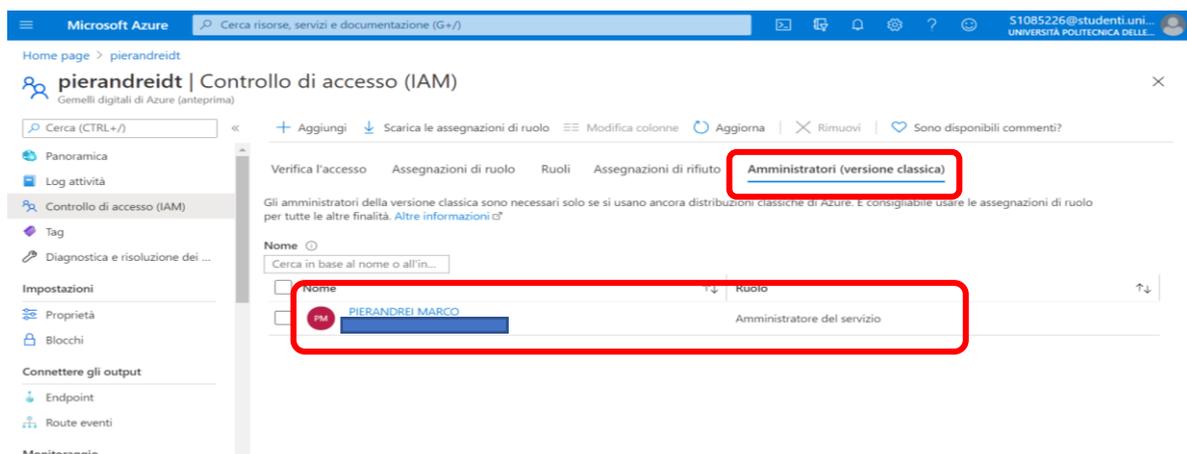
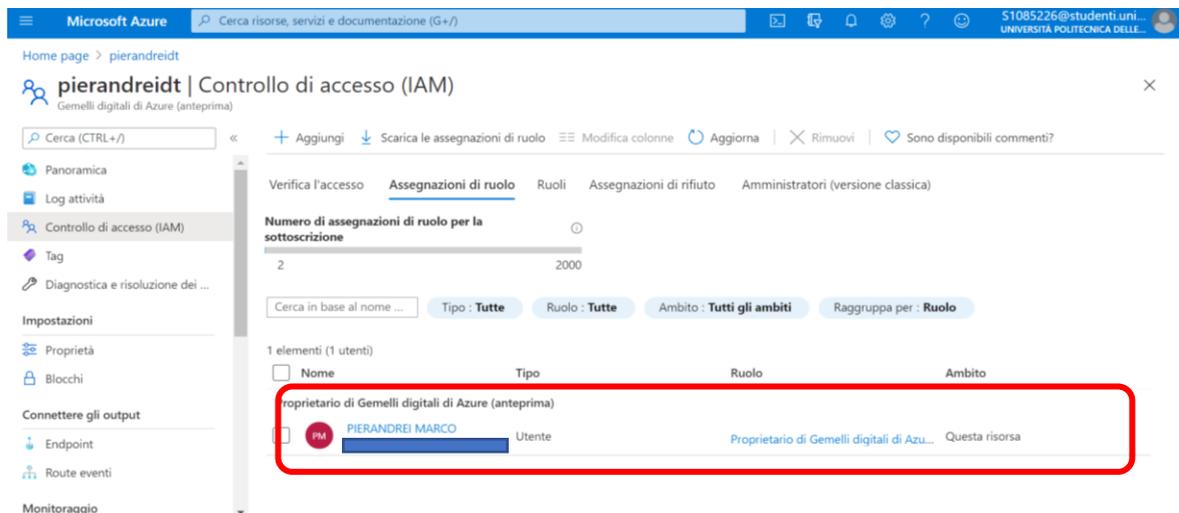


Fig.14,15

Una volta eseguito lo script della creazione dell'istanza, si creerà anche una app relativa ad essa da cui dovremo attingere dati importantissimi per il corretto proseguimento della creazione di Gemelli Digitali di Azure. Di seguito verranno illustrati questi dati dell'app, facilmente reperibili nel portale di Azure.

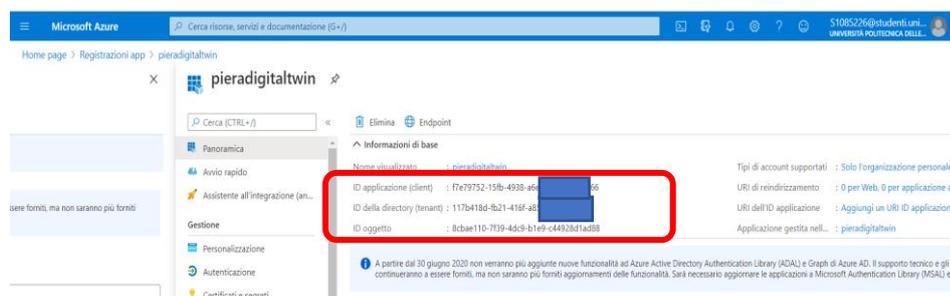


Fig.16

## 2.6 Creazione di una Applicazione Client

Quando si sviluppano Digital Twin con Azure è consigliabile, scrivere un'applicazione client. Per definizione in informatica un client rappresenta un qualunque componente software che risiede su una qualsiasi macchina chiamata host e che può accedere a tutti quelli che sono i servizi e le risorse di un'altra componente chiamata server seguendo logicamente dei protocolli di comunicazione specifici; in seguito a questa definizione riusciamo a capire meglio il perché necessitiamo di questo tipo di applicazione, perché essa ci consenta di interagire con la nostra istanza di servizio.

I due passaggi che sono stati chiamati a svolgere sono stati:

- configurare il progetto
- scrivere il codice dello stesso.

Per la realizzazione del primo obiettivo, è stato necessario l'utilizzo del prompt dei comandi e l'esecuzione di tre comandi il primo per la creazione di un progetto C# per la console ed i restanti due necessari per l'uso dei Gemelli Digitali uno per la *libreria client di Gemelli Digitali IoT di Azure* e l'altro per eseguire l'autenticazione sempre in Azure.

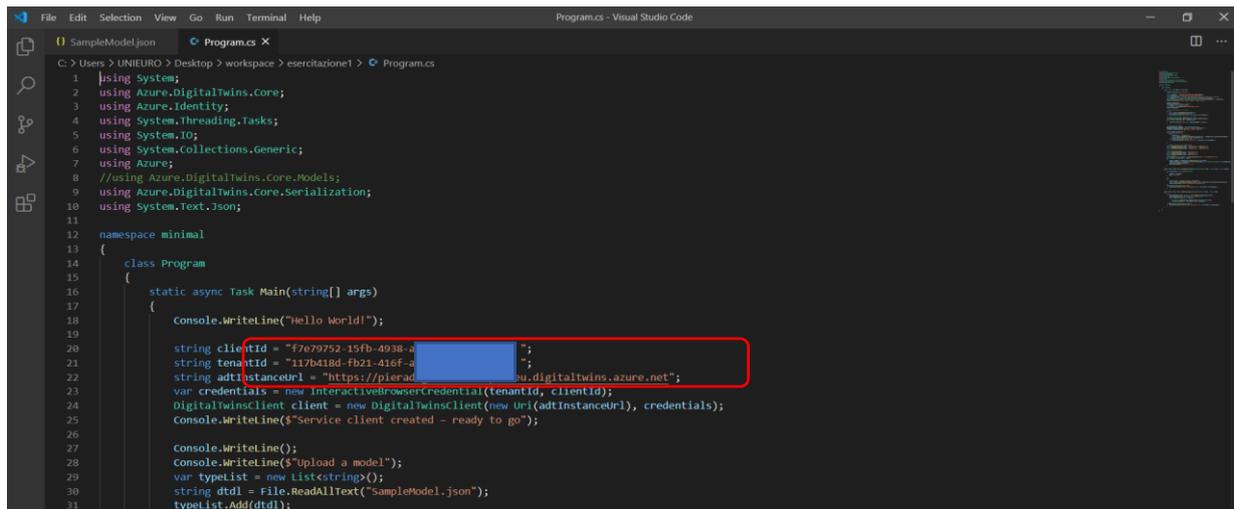
```
dotnet new console
```

```
dotnet add package Azure.DigitalTwins.Core --version 1.0.0-preview.3  
dotnet add package Azure.identity
```

Mentre per quanto riguarda la scrittura del codice dell'app per l'uso di Gemelli Digitali di Azure ci sono diversi passaggi, verranno tutti elencati e mostrati passo dopo passo per comprendere a meglio le azioni fatte. È stato inoltre necessario installare sul pc un programma chiamato Visual Studio Code che mi ha permesso di scrivere il codice e salvarlo nei formati richiesti.

### 2.6.1 Autenticazione per il servizio

Per l'autenticazione del servizio sono necessari quelle informazioni che nei paragrafi precedenti erano state riquadrate nelle immagini, come il TenantId, ClientId, adInstanceUrl, quindi le prime righe di codice che abbiamo scritto sono le seguenti



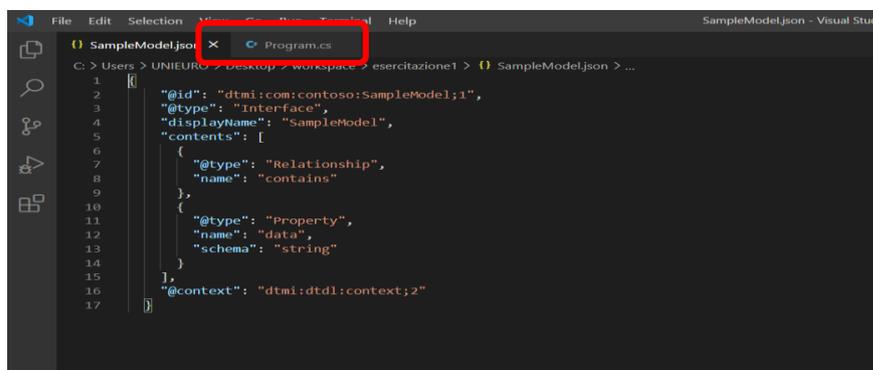
```
1 using System;
2 using Azure.DigitalTwins.Core;
3 using Azure.Identity;
4 using System.Threading.Tasks;
5 using System.IO;
6 using System.Collections.Generic;
7 using Azure;
8 //using Azure.DigitalTwins.Core.Models;
9 using Azure.DigitalTwins.Core.Serialization;
10 using System.Text.Json;
11
12 namespace minimal
13 {
14     class Program
15     {
16         static async Task Main(string[] args)
17         {
18             Console.WriteLine("Hello World!");
19
20             string clientId = "f7e79752-15fb-4938-a";
21             string tenantId = "117b418d-fb21-416f-a";
22             string adInstanceUrl = "https://plera.azure.digitaltwins.azure.net";
23             var credentials = new InteractiveBrowserCredential(tenantId, clientId);
24             DigitalTwinsClient client = new DigitalTwinsClient(new Uri(adInstanceUrl), credentials);
25             Console.WriteLine($"Service client created - ready to go!");
26
27             Console.WriteLine();
28             Console.WriteLine($"Upload a model");
29             var typeList = new List<string>();
30             string dtId = File.ReadAllText("SampleModel.json");
31             typeList.Add(dtId);
```

Fig.17

una volta che scritte queste righe di codice si salva il tutto ed utilizzando la finestra di comando si esegue il comando run. Da notare che nelle prime nove righe del codice, come mostrato nella figura sovrastante, sono presenti degli using che fanno riferimento anche a parti successive del codice completo.

## 2.6.2 Caricare un modello

Come già detto Azure Digital Twin non include un vocabolario di dominio intrinseco, quindi i tipi di elementi dell'ambiente che è possibile realizzare in questa risorsa di Azure, vengono specificati dal programmatore utilizzando i modelli. Nel mio caso il modello per i gemelli digitali ci è stato suggerito dalla guida di Azure, quindi una volta analizzato e studiato lo abbiamo inserito all'interno di un nuovo documento sempre all'interno di Visual Studio Code, denominato SampleModel.json.



```
1 {
2   "@id": "dtmi:com:contoso:SampleModel;1",
3   "@type": "Interface",
4   "displayName": "SampleModel",
5   "contents": [
6     {
7       "@type": "Relationship",
8       "name": "contains"
9     },
10    {
11      "@type": "Property",
12      "name": "data",
13      "schema": "string"
14    }
15  ],
16  "@context": "dtmi:dtId1:context;2"
17 }
```

Fig.18

Nell'altro file dove viene scritto il codice completo, abbiamo dovuto aggiungere un frammento di codice che ci consente di caricare il file DTDL di cui abbiamo parlato sopra, che è stato creato sul disco, lo prende e lo carica in Azure Digital Twin. In oltre abbiamo aggiunto anche un'istruzione di stampa che ci aiutasse a capire se i modelli sono stati caricati correttamente o meno. Al fine di evitare che si verificano errori, abbiamo aggiunto anche le funzioni **try** e **catch**.

```
32
33 // Upload the model to the service
34 try {
35     await client.CreateModelsAsync(typeList);
36 } catch (RequestFailedException rex) {
37     Console.WriteLine($"Load model: {rex.Status}:{rex.Message}");
38 }
39 // Read a list of models back from the service
40 AsyncPageable<ModelData> modelDataList = client.GetModelsAsync();
41 await foreach (ModelData md in modelDataList)
42 {
43     Console.WriteLine($"Type name: {md.DisplayName}: {md.Id}");
44 }
45
```

Fig.19

### 2.6.3 Creazione dei Gemelli Digitali

Dopo che ho caricato il modello dei gemelli digitali, è stato necessario aggiungere la funzione che mi permettesse di creare questi gemelli. Per definizione i gemelli sono delle istanze e rappresentano le entità all'interno dell'ambiente in esame, nel mio caso erano le stanze dell'edificio ed i piani.

```
45
46 // Initialize twin data
47 BasicDigitalTwin twinData = new BasicDigitalTwin();
48 twinData.Metadata.ModelId = "dtmi:com:contoso:SampleModel;1";
49 twinData.CustomProperties.Add("data", $"Hello World!");
50
51 string prefix="sampleTwin-";
52 for(int i=0; i<5; i++) {
53     try {
54         twinData.Id = $"{prefix}{i}";
55         await client.CreateDigitalTwinAsync($"{prefix}{i}", JsonSerializer.Serialize(twinData));
56         Console.WriteLine($"Created twin: {prefix}{i}");
57     } catch (RequestFailedException rex) {
58         Console.WriteLine($"Create twin error: {rex.Status}:{rex.Message}");
59     }
60 }
61
```

Fig.20

Quindi guardando l'immagine sopra e analizzando il codice si può notare come alla riga numero 48 si faccia riferimento al modello SampleModel che avevamo caricato e di cui avevamo parlato in precedenza, poi scendendo alla riga numero 52 possiamo vedere che abbiamo creato cinque gemelli dal 0 al 4.

## 2.6.4 Creazione delle relazioni

Dopo aver creato questi cinque Gemelli Digitali è necessario organizzarli tra loro tramite delle relazioni, e per connetterli all'interno di un grafo. I grafi sono utilizzati per riuscire ad avere una rappresentazione dell'intero ambiente. Queste relazioni vengono fondamentalmente scelte dal tipo di lavoro che ogni gemello svolge e con quale altro deve interagire per riuscire a raggiungere l'obiettivo finale. Quindi ho scritto un pezzo di codice che andasse a definire queste relazioni ed altre righe per poter visualizzare l'elenco di tutte le relazioni.

```
81
82     public async static Task CreateRelationship(DigitalTwinsClient client, string srcId, string targetId)
83     {
84         var relationship = new BasicRelationship
85         {
86             TargetId = targetId,
87             Name = "contains"
88         };
89
90         try
91         {
92             string relId = $"{srcId}-contains->{targetId}";
93             await client.CreateRelationshipAsync(srcId, relId, JsonSerializer.Serialize(relationship));
94             Console.WriteLine("Created relationship successfully");
95         }
96         catch (RequestFailedException rex) {
97             Console.WriteLine($"Create relationship error: {rex.Status}:{rex.Message}");
98         }
99     }
100
101     public async static Task ListRelationships(DigitalTwinsClient client, string srcId)
102     {
103         try {
104             AsyncPageable<string> results = client.GetRelationshipsAsync(srcId);
105             Console.WriteLine($"Twin {srcId} is connected to:");
106             await foreach (string rel in results)
107             {
108                 var brel = JsonSerializer.Deserialize<BasicRelationship>(rel);
109                 Console.WriteLine($" -{brel.Name}->{brel.TargetId}");
110             }
111         } catch (RequestFailedException rex) {
112             Console.WriteLine($"Relationship retrieval error: {rex.Status}:{rex.Message}");
113         }
114     }
115 }
116 }
117 }
```

Fig.21

```
61
62     // Connect the twins with relationships
63     await CreateRelationship(client, "sampleTwin-0", "sampleTwin-1");
64     await CreateRelationship(client, "sampleTwin-0", "sampleTwin-2");
65     await CreateRelationship(client, "sampleTwin-2", "sampleTwin-3");
66
67     //List the relationships
68     await ListRelationships(client, "sampleTwin-0");
69     await ListRelationships(client, "sampleTwin-2");
70     await ListRelationships(client, "sampleTwin-4");
71     // Run the program
```

Fig.22

Dalla figura mostrata sopra è possibile evincere le relazioni che ho deciso io di dare ai vari gemelli, quindi esisterà una relazione tra 0-1, 0-2, 2-3.

## 2.6.5 Query sui Gemelli Digitali

Una delle funzionalità principali dei Digital Twins di Azure è proprio la possibilità di eseguire delle query, cioè interrogare il sistema per avere delle informazioni in maniera semplice e veloce sul sistema in esame. Nel mio caso quindi ho aggiunto questa funzione per avere il codice completo ed operativo.

```
71 // Run a query
72 AsyncPageable<string> result = client.QueryAsync("Select * From DigitalTwins");
73 await foreach (string twin in result)
74 {
75     object jsonObj = JsonSerializer.Deserialize<object>(twin);
76     string prettyTwin = JsonSerializer.Serialize(jsonObj, new JsonSerializerOptions { WriteIndented = true });
77     Console.WriteLine(prettyTwin);
78     Console.WriteLine("-----");
79 }
80 }
81
```

**Fig.23**

## 2.7 Approfondire i Gemelli Digitali con un'app client

Dopo aver creato l'applicazione client ora parleremo di come questa funzione di Azure può essere utilizzata per l'interazione con i Gemelli Digitali. Questo lavoro ci è servito per capire meglio delle azioni essenziali dei Digital Twin che anche se erano state già viste nei passi precedenti qui sono state approfondite, per elencarne qualcuna ad esempio, il caricamento dei modelli, creazione e modifica dei gemelli e relative relazioni. In questo approfondimento ci è stato possibile esaminare il codice ed andare a ricavarci alcune informazioni importanti qual ora un domani si volesse modificare il progetto base e adattarlo alla nuova esigenza. Per fare tutto questo è stato necessario scaricare un programma chiamato Visual Studio 2019, inoltre seguendo la guida di Azure consigliava di scaricare un file al cui interno era presente un progetto di esempio scritto in C# e seguire quel progetto li. I primi passaggi sono stati quelli specifici per la configurazione del progetto, nel quale dovevamo inserire sempre i dati fondamentali che ci portiamo dietro dall'inizio, cioè i dati dell'istanza e gli ID tenant e client. Una volta inseriti tutti i dati principali si prosegue con le successive fasi per il completamento del compito.

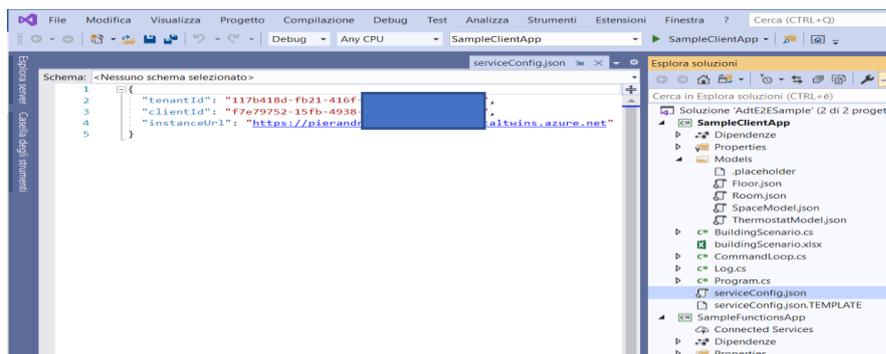
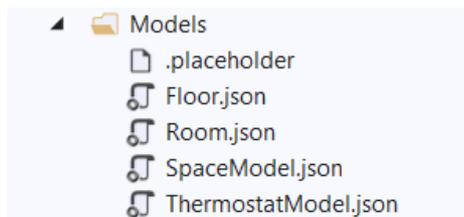


Fig.24

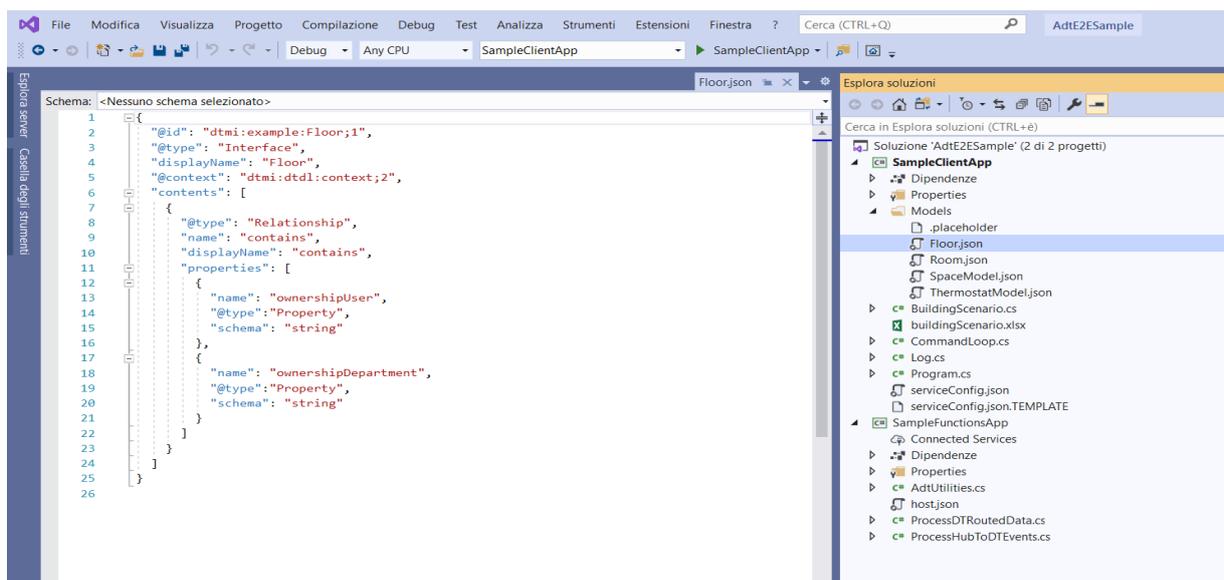
### 2.7.1 Modellare un ambiente fisico

Come è stato già detto, la prima cosa da fare per la creazione dei Gemelli Digitali è quella di andare a definire e progettare quelli che sono i modelli. Aprendo il file su Visual Studio abbiamo una sezione dedicata solo ai modelli, alcuni dei quali sono stati già visti precedentemente quando abbiamo dato le definizioni degli stessi. Ora però analizzeremo il processo di come sono stati inseriti e modificati.



**Fig.25**

Questi elencati nell'immagine qui sopra sono tutti i modelli da noi utilizzati, siamo partiti da Room.json al quale abbiamo modificato il numero della versione, in maniera tale da utilizzare una versione più aggiornata; abbiamo inserito la proprietà Humiditylevel, Roomname e la relazione contains per far sì che questo tipo di gemello formi delle relazioni di quel genere con altri gemelli. Per quanto riguarda gli altri modelli, erano già presenti all'interno del file e non richiedevano nessuna modifica, quindi il nostro compito è stato quello di analizzare il codice, verificare che tutto quello che apparteneva a quel modello era corretto e faceva al nostro caso, ed una volta controllato il tutto, abbiamo continuato con i passaggi successivi. Di seguito è possibile visionare il modello Floor.json, un modello di tipo interfaccia, molto semplice caratterizzato da una relazione di tipo contains e da due proprietà.

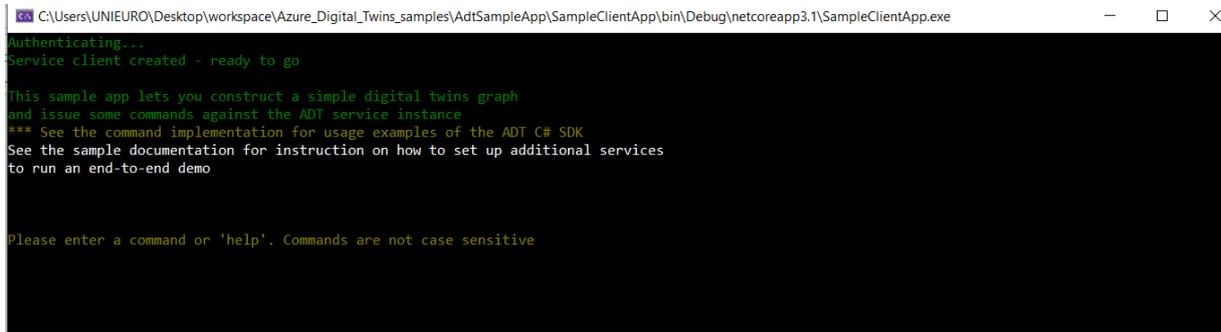


**Fig.26**

## 2.7.2 Avvio dell'app

Dopo che abbiamo definito tutti i modelli, quello che è stato necessario fare è di utilizzare l'app per integrare con l'istanza dei Digital Twin, che fu la prima cosa che creammo. Per fare questa integrazione è quindi necessario eseguire il progetto, cliccando con il mouse sopra questo pulsante , avvieremo il progetto. Successivamente si aprirà una finestra di comando in cui dovremo effettuare l'autenticazione, questa viene gestita tramite browser, dovremo

solamente inserire le credenziali una volta che quest'ultimo passaggio va a buon fine sul command apparirà le seguenti scritte.



```
C:\Users\UNIEURO\Desktop\workspace\Azure_Digital_Twins_samples\AdtSampleApp\SampleClientApp\bin\Debug\netcoreapp3.1\SampleClientApp.exe
Authenticating...
Service client created - ready to go

This sample app lets you construct a simple digital twins graph
and issue some commands against the ADT service instance
*** See the command implementation for usage examples of the ADT C# SDK
See the sample documentation for instruction on how to set up additional services
to run an end-to-end demo

Please enter a command or 'help'. Commands are not case sensitive
```

**Fig.27**

### 2.7.3 Caricamento modelli

Dopo che li abbiamo progettati, è necessario caricare i modelli nell'istanza Digital Twin Azure. Questo passaggio consente di configurare l'istanza del servizio Gemelli Digitali con il glossario del dominio specifico. Dopo aver caricato i modelli vi è la fase di creazione delle istanze di gemelli che li impiegano. Si procederà con l'inserimento all'interno del pannello di controllo dei comandi per caricare il modello Room.json nella nuova versione e per caricare il modello Floor.json il quale ci consente di creare tipi diversi di gemelli. Tramite il comando GetModels true è possibile verificare il corretto caricamento dei modelli, questo comando esegue una query sull'istanza, in maniera tale da recuperare tutti i modelli caricati verranno stampate tutte le informazioni relative ad essi in maniera completa.

Nell'immagine sottostante è possibile vedere cerchiato in rosso il comando e poi a seguire il primo modello relativo alla Room e via di seguito tutti gli altri, in basso nell'immagine riquadrato in rosso è possibile notare il modello Floor.

```
C:\Users\UNIEURO\Desktop\workspace\Azure_Digital_Twins_samples\AdtSampleApp\SampleClient
Please enter a command or 'help'. Commands are not case sensitive
GetModels true
Submitting:
dtmi:example:Room;2
Response:
{
  "@id": "dtmi:example:Room;2",
  "@type": "Interface",
  "displayName": "Room",
  "contents": [
    {
      "@type": "Property",
      "name": "Temperature",
      "schema": "double"
    },
    {
      "@type": "Property",
      "name": "HumidityLevel",
      "schema": "double"
    },
    {
      "@type": "Property",
      "name": "RoomName",
      "schema": "string"
    },
    {
      "@type": "Relationship",
      "name": "contains"
    }
  ],
  "@context": [
    "dtmi:dtdl:context;2"
  ]
}
dtmi:example:Floor;1
Response:
{
  "@id": "dtmi:example:Floor;1",
```

Fig.28

## 2.7.4 Errori

Per quanto riguarda gli errori, la nostra applicazione è in grado di gestire anche gli errori di servizio. Infatti se noi proviamo ad eseguire nuovamente un comando come ad esempio quello di *CreateModels Room* il sistema ci stamperà l'errore 409 e ci dirà che è presente un conflitto. Il conflitto si verifica a causa della presenza già di un modello Room e per questo motivo non è possibile sovrascrivere modelli. Qualora si verificassero altri problemi Microsoft Azure dispone di una guida con i relativi errori e come correggerli. Sempre in questa esercitazione durante la fase di run è stato stampato l'errore 403, con il quale veniva negato l'accesso relativo alle richieste di servizio. Le cause di questo errore erano dovute al fatto che utilizzando un'app client per poter comunicare con i gemelli digitali, la registrazione dell'app, di cui abbiamo parlato nei capitoli precedenti, non aveva le corrette autorizzazioni impostate per il servizio Azure Digital Twin. Quindi la guida consiglia di fare delle verifiche proprio in tema di autorizzazioni e modificarle, inserendo quelle corrette, nel nostro caso era una questione di ruoli di utenza Azure, non era stato impostato il ruolo di proprietario (anteprima) di Azure Digital Twin. Altro tipo di errore che potrebbe comparire è il numero 400 (bad request) il quale sta ad indicare un comando scritto nella maniera sbagliata, e quindi il sistema ci invita a riscrivere il comando correggendo gli errori.

```
C:\Users\UNIEURO\Desktop\workspace\Azure_Digital_Twins_samples\AdtSampleApp\SampleClientApp\bin\Debug\netcoreapp3.1\SampleClientApp.exe
Please enter a command or 'help'. Commands are not case sensitive
createModels Room
loading from C:\Users\UNIEURO\Desktop\workspace\Azure_Digital_Twins_samples\AdtSampleApp\SampleClientApp\bin\Debug\netcoreapp3.1\Models
submitting models: Room.json...
response 409: Service request failed.
status: 409 (Conflict)
content:
  "error": { "code": "ModelAlreadyExists", "message": "Model with same ID already exists dtmi:example:Room;2. Use Model_list API to view models that already exist. See the Swagger example. (http://aka.ms/ModelListSwagger)" }
headers:
  Strict-Transport-Security: REDACTED
  Date: Mon, 28 Sep 2020 09:26:51 GMT
  Content-Length: 216
  Content-Type: application/json; charset=utf-8
Please enter a command or 'help'. Commands are not case sensitive
```

Fig.29

## 2.7.5 Creazione dei Gemelli Digitali

Dopo che i modelli sono stati caricati nell'istanza, è stato possibile creare i gemelli digitali in base a quelle che erano le definizioni dei modelli. Per la creazione di un gemello si utilizza nel pannello di controllo il comando *CreateDigitalTwin*, è necessario però riferirsi ad un modello sul quale poi si baserà il Gemello, inoltre, questa cosa è facoltativa, si possono andare a definire quelli che sono tutti i valori iniziali delle proprietà appartenenti al modello nel caso specifico sono stati impostati valori di temperatura pari a 70 e di umidità pari a 30 per la *Room0* mentre per la *Room1*. In questo passaggio non è stato necessario andare a definire o includere quelle che sono le relazioni. Nel pannello di controllo compariranno tutti i gemelli creati con successo.

```
C:\Users\UNIEURO\Desktop\workspace\Azure_Digital_Twins_samples\AdtSampleApp\SampleClientApp\bin\Debug\netcoreapp3.1\SampleClientApp.exe
Please enter a command or 'help'. Commands are not case sensitive
CreateDigitalTwin dtmi:example:Room;2 room0 RoomName string Room0 Temperature double 70 HumidityLevel double 30
Preparing...
Submitting...
Twin 'room0' created successfully
Please enter a command or 'help'. Commands are not case sensitive
CreateDigitalTwin dtmi:example:Room;2 room1 RoomName string Room1 Temperature double 80 HumidityLevel double 60
Preparing...
Submitting...
Twin 'room1' created successfully
Please enter a command or 'help'. Commands are not case sensitive
CreateDigitalTwin dtmi:example:Floor;1 floor0
Preparing...
Submitting...
Twin 'floor0' created successfully
Please enter a command or 'help'. Commands are not case sensitive
CreateDigitalTwin dtmi:example:Floor;1 floor1
Preparing...
Submitting...
Twin 'floor1' created successfully
Please enter a command or 'help'. Commands are not case sensitive
```

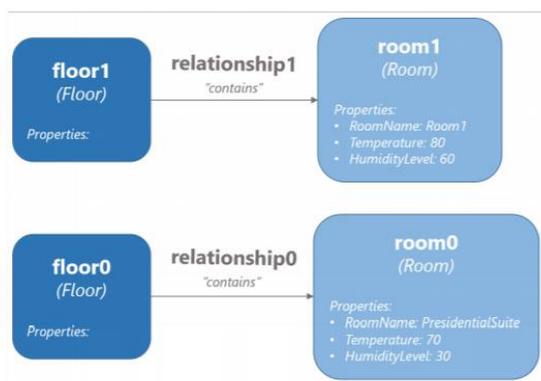
Fig.30

Nel caso che possiamo vedere sopra i gemelli creati sono *Room0*, *Room1*, *Floor0* e *Floor1*.

Se si intende cambiare le proprietà ad un gemello è possibile farlo attraverso l'utilizzo di un comando *UpdateDigitalTwin* poi si sceglie la proprietà che si vuole modificare ed in quale gemello si intende farlo successivamente si aggiungono i dati o i nomi da modificare.

## 2.7.6 Creazione grafo e specifica relazioni

Una volta che sono stati creati i gemelli è stato possibile creare delle relazioni tra di essi per poi successivamente poterli connettere in un grafo. Come abbiamo già specificato i grafi servono per poter rappresentare l'ambiente per intero. Sempre sul command è stato possibile creare queste relazioni, grazie all'utilizzo della funzione *CreateRelationship*. Per prima cosa si è dovuto specificare da quale gemello provenisse la relazione, poi il tipo di relazione ed in fine a quale gemello fosse destinata per la connessione. Successivamente a questi tre passaggi fondamentali si è dovuto scegliere un nome per identificare la relazione. Nel nostro caso studio abbiamo creato questo tipo di relazioni.



**Fig.31**

Successivamente alla creazione del grafo e delle relazioni, è possibile eseguire delle query su di esso, in maniera molto semplice che ci consente di dare una risposta a ciò che cerchiamo sull'ambiente. Quindi inserendo dei comandi nel pannello di controllo come *Query* si riesce ad avere una panoramica di tutti quelli che sono i gemelli presenti. Possono essere fatte ad esempio delle domande più specifiche, con richieste dettagliate su quello che quel dato gemello rappresenta.

## 2.8 Soluzione end-to-end

In questa fase è necessario configurare una soluzione end-to-end, basta sull'utilizzo di dati rilevati in tempo reale dall'ambiente sottoposto in esame, quindi è necessario connettere l'istanza che è stata citata più volte nei capitoli precedenti per la gestione di dispositivi e dati. I passaggi per la configurazione non sono molti ma un pochino più complessi rispetto ai precedenti, è stato necessario creare un'istanza per i componenti precedentemente creati, utilizzare un'app facente parte delle Funzioni Azure, che consente di instradare i dati di telemetria preventivamente simulati nel dispositivo Hub IoT verso le proprietà dei gemelli da noi creati ed in fine diffondere le modifiche attraverso il Grafo dei Gemelli formulando le notifiche dei gemelli con delle apposite Funzioni di Azure come Endpoint e Route. Per lo sviluppo completo di questi passaggi è stato richiesto l'utilizzo del Cloud Shell appropriatamente configurato ed aggiornato con le corrette estensioni che ci consentono di abilitare i comandi per poter interagire con i nostri Gemelli Digitali. In secondo luogo, come per l'approfondimento dell'app client è necessario utilizzare lo stesso file con già la configurazione adottata e salvata. Come detto nell'introduzione il nostro caso studio comprende un edificio con piani e stanze nelle quali sono inseriti dei sensori, il tutto sarà rappresentato digitalmente. Di seguito sono illustrati i passaggi compiuti con relativo schema di rappresentazione.

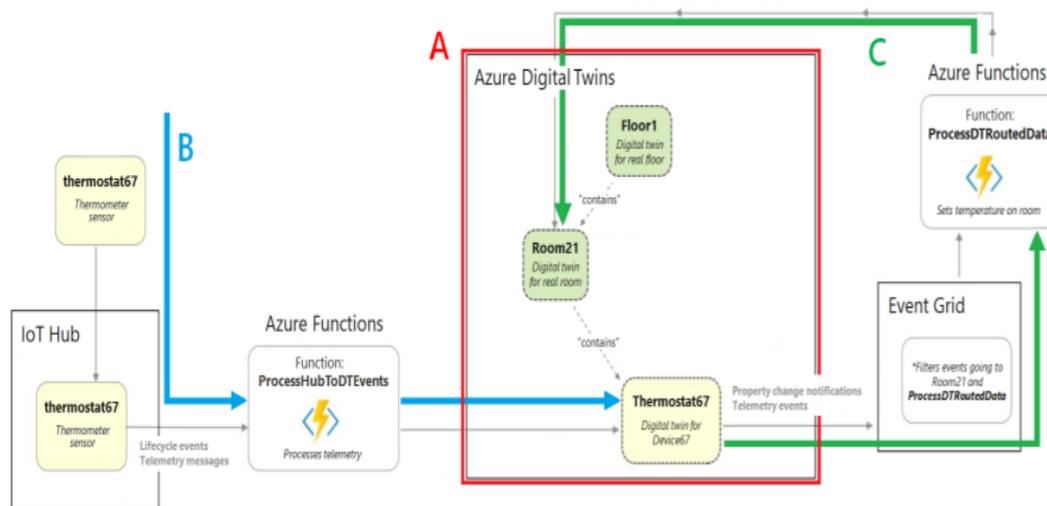
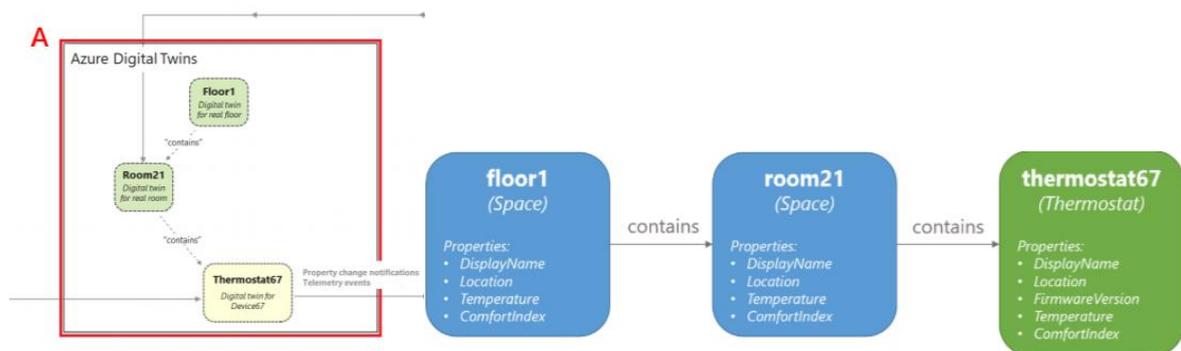


Fig.32

## 2.8.1 Creazione dell'istanza

Sempre sul file che abbiamo scaricato, aprendo la soluzione *AdtSampleApp*, è stato possibile creare il componente Gemelli Digitali dello scenario end-to-end. Mandandolo in esecuzione, verrà chiesta come in precedenza l'autenticazione sul browser, una volta concluso questo passaggio, il pannello di controllo è pronto a ricevere i comandi. Infatti si è proceduto con l'inserimento del comando *SetupBuildingScenario*, questo comando implica la creazione dei tre soggetti esistenti nell'ambiente reale all'interno dell'istanza; un piano *Floor1*, una stanza *Room21* ed il sensore di temperatura *Thermostat67*. Una volta eseguito sono stati inviati dei messaggi di conferma di corretta esecuzione. Il grafo che consente di capire il collegamento tra i tre soggetti principali di questa operazione. Per avere una maggiore conferma è possibile eseguire una Query la quale ci mostrerà quali gemelli sono stati creati.



**Fig.33**

## 2.8.2 Configurare l'app

Successivamente è stato necessario configurare l'app per le Funzioni di Azure, la quale è stata di aiuto per l'elaborazione dei dati. Questa applicazione denominata *SampleFunctionApp* nello specifico svolge due funzioni, una quella di elaborare dati dell'Hub IoT in ingresso aggiornando l'istanza dei Gemelli e l'altra quella di elaborare i dati in uscita dai Gemelli, aggiornando i Gemelli padre nell'istanza. Queste due operazioni si chiamano *ProcessHubToDTEvents* & *ProcessDTRoutedData*. Durante lo svolgimento di questi compiti si è reso necessario pubblicare l'app per le funzioni, in maniera tale da verificare che essa fosse in grado di accedere ai Gemelli, tramite l'assegnazione di una identità di Azure Active Directory (Azure AD). I passaggi per la pubblicazione si sono svolti tutti tramite Visual Studio 2019, per prima sono state aggiornate le dipendenze alle versioni più recenti, una volta completato questo passaggio è stata pubblicata l'app assegnandogli il gruppo di risorse che avevamo già visto nel primo paragrafo, l'istanza e la sottoscrizione/licenza a nostra disposizione di Azure. Una volta

pubblicata si è convenuto di assegnare le autorizzazioni all'app per le funzioni, i passaggi sono stati i seguenti:

- Configurazione di una impostazione dell'app
- Assegnazione di una identità Azure AD all'applicazione in maniera tale che sia gestita dal sistema
- Assegnare il ruolo di *Proprietario di Gemelli Digitali di Azure (Anteprima)* all'identità precedentemente creata, tutto questo nell'istanza dei Gemelli

Per poter svolgere tutti questi passaggi, era stato citato all'inizio, è stato utilizzato il Cloud Shell, all'interno del quale sono stati inseriti dei comandi ed è stato possibile creare il tutto.

Nella immagine sottostante è ritratta la nostra applicazione pubblicata tramite il programma Visual Studio 2019.

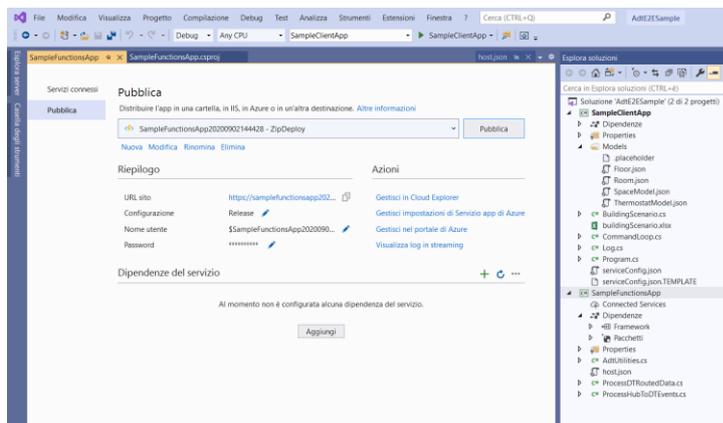
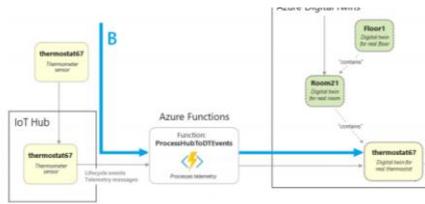


Fig.34

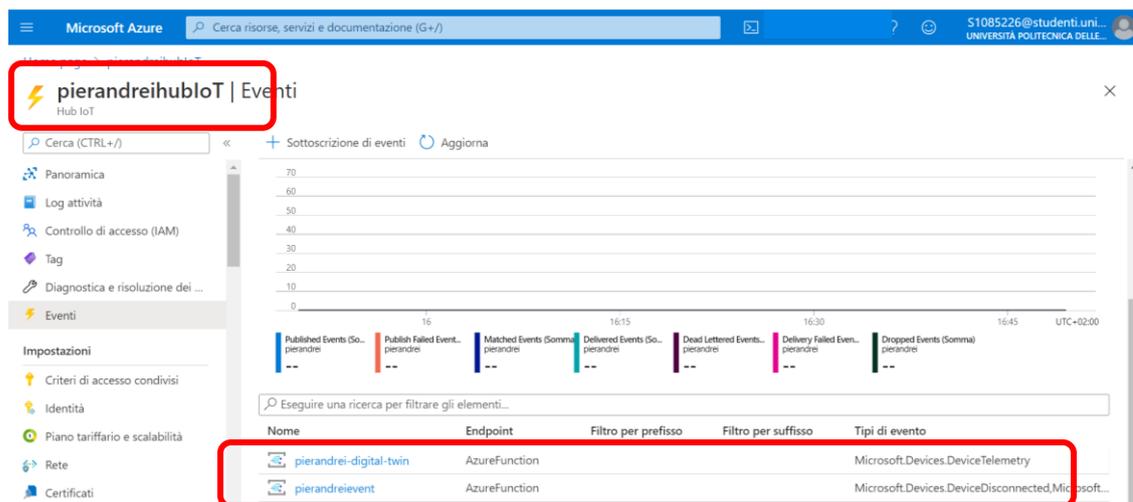
### 2.8.3 Elaborazione dei dati simulati da un dispositivo Hub IoT

In generale il grafo dei Gemelli è stato progettato per essere basato sui dati di telemetria di dispositivi reali. Nel caso studio quindi il termostato simulato, registrato nell'Hub IoT, è stato connesso al gemello digitale che lo rappresenta nell'istanza Gemelli Digitali di Azure. I dati che esso simula, vengono orientati tramite quella funzione che è stata nominata nel paragrafo precedente *ProcessHubToDTEvents*, la quale attiva un aggiornamento nel corrispondente Gemello così è possibile avere il DT sempre aggiornato. Questo fluire di dati convogliati è chiamato ROUTING di EVENTI.



**Fig.35**

Per prima cosa è stata creato tramite il Cloud Shell l'istanza dell'Hub IoT, perché i Gemelli Azure sono progettati per lavorare insieme ad essa; questo Hub gestirà tutto il dispositivo. Una volta creato si è dovuto connettere l'Hub alla funzione Azure, in maniera tale che i dati potessero essere trasmessi dal dispositivo allo stesso Hub tramite la funzione che aggiorna i Gemelli. Per poter connettere il tutto si è dovuto creare una sottoscrizione di eventi nell'Hub in maniera tale che la funzione di Azure funzionasse da endpoint. In questo modo si sottoscrive la funzione agli eventi che si verificano nell'Hub IoT.



**Fig.36**

Successivamente alla creazione dell'Hub si è proceduto con la registrazione del dispositivo di telemetria simulato con lo stesso Hub. Quindi è stato creato una rappresentazione del dispositivo nell'Hub con l'ID corrispondente thermostat67; quest'ultimo si è connesso alla rappresentazione sopra indicata ed è stato possibile avere il trasferimento degli elementi di telemetria dal dispositivo all'Hub, in cui la funzione Azure rimane connessa e può continuare ad aggiornare. L'ultimo passaggio è stato quello di configurare il simulatore per poter inviare i dati all'istanza dell'Hub IoT.

La simulazione ci stampa questi valori:

```
C:\Users\UNIEURO\Desktop\workspace\Azure_Digital_Twins_samples\DeviceSimulator\DeviceSimulator\bin\Debug\netcoreapp3.1\...
Press CTRL+C to exit
30/09/2020 17:37:25 > Sending message: {"Temperature":67.9572626859682}
Message received. Partition: 3 Data: '{"Temperature":67.9572626859682}'
30/09/2020 17:37:30 > Sending message: {"Temperature":68.48416728967995}
Message received. Partition: 3 Data: '{"Temperature":68.48416728967995}'
30/09/2020 17:37:35 > Sending message: {"Temperature":67.0782309882707}
Message received. Partition: 3 Data: '{"Temperature":67.0782309882707}'
30/09/2020 17:37:40 > Sending message: {"Temperature":69.39983843378715}
Message received. Partition: 3 Data: '{"Temperature":69.39983843378715}'
30/09/2020 17:37:45 > Sending message: {"Temperature":69.3036564599274}
Message received. Partition: 3 Data: '{"Temperature":69.3036564599274}'
30/09/2020 17:37:50 > Sending message: {"Temperature":69.20941074667937}
Message received. Partition: 3 Data: '{"Temperature":69.20941074667937}'
30/09/2020 17:37:55 > Sending message: {"Temperature":70.7898138276254}
Message received. Partition: 3 Data: '{"Temperature":70.7898138276254}'
```

Fig.37

### 2.8.4 Propagazione eventi di Gemelli Digitali tramite il grafo

Fino a questo momento si è parlato solamente di come aggiornare i Gemelli di Azure da dati di dispositivi esterni. E' stato deciso inoltre di affrontare l'argomento della propagazione delle modifiche apportate ad un gemello attraverso il grafo dei Gemelli Digitali di Azure. Per svolgere questo compito è stato necessario l'impiego della funzione che avevamo elencato al paragrafo [2.7.2], *ProcessDTRoutedData*, quindi questa funzione ci aggiorna il gemello Room quando il gemello Thermostat connesso, viene aggiornato.

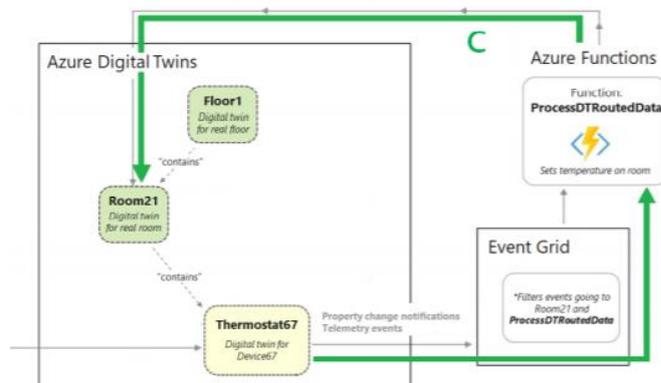


Fig.38

Per riuscire a portare a termine questo task di configurazione del flusso dati, si sono dovuti svolgere dei passaggi, che saranno elencati di seguito.

Il primo di questi passaggi fondamentali è stato, la configurazione dell'endpoint. Tramite la Griglia Eventi, che è un servizio di Azure il quale ci consente di convogliare e consegnare, tutti quelli che sono gli eventi provenienti da un qualsiasi servizio Azure, verso altri posti sempre all'interno di Azure. Inoltre è stato necessario creare un Argomento di Griglia degli Eventi, il cui compito è quello di ricavare determinati eventi da un'origine, in questo caso è possibile per i sottoscrittori, rimanere in ascolto sull'argomento in maniera tale che possono acquisire gli

eventi non appena vengono inviati. Tramite Cloud Shell di Azure è stato possibile creare l'argomento e il relativo endpoint al suo interno.

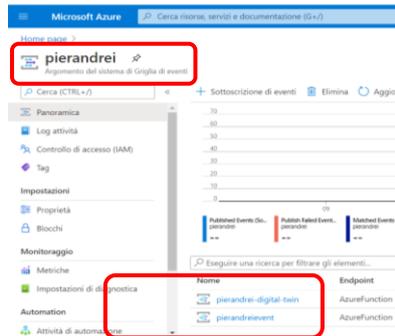


Fig.39

Il passo successivo è stato quello di creare una route di Gemelli Digitali, la quale potesse inviare gli eventi all'endpoint precedentemente creato. Il comando da inserire nel Cloud Shell per creare la route è il seguente `az dt route create --dt-name pierandreimarcodigitaltwin --endpoint-name pierandreievent --route-name pierandreiroute`.

Di seguito si è proceduto con la connessione della funzione alla Griglia Eventi, ricordando la funzione prima citata *ProcessDTRoutedData*, questa è stata sottoscritta all'argomento Griglia degli eventi, con lo scopo di trasferire i dati di telemetria dal gemello *Thermostat67*, attraverso l'argomento, direttamente alla funzione; la quale torna ai Gemelli Digitali di Azure ed aggiorna il gemello della *Room21*. A tale scopo si è ritenuto necessario creare una sottoscrizione di Griglia degli Eventi alla funzione di Azure *ProcessDTRoutedData* come endpoint.

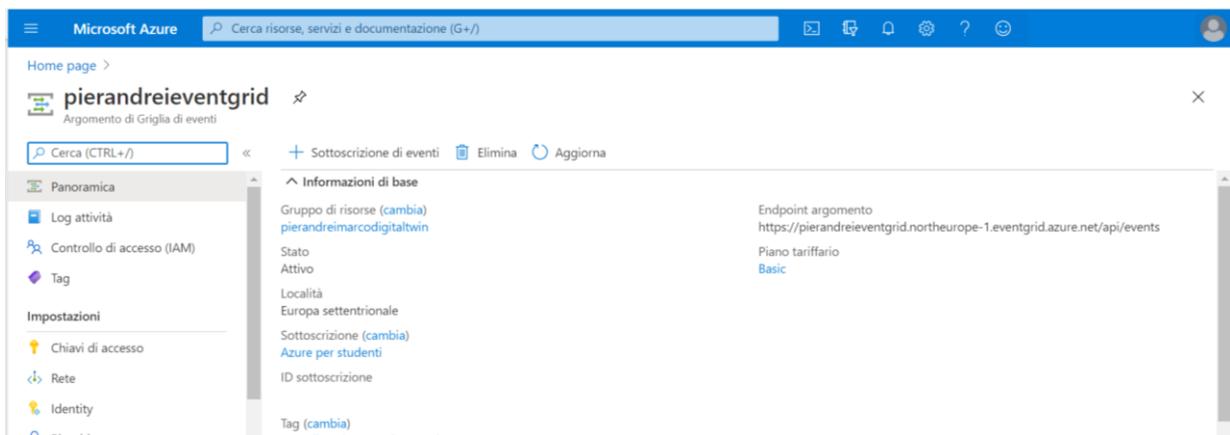


Fig.40

## 3. Sviluppi futuri

### 3.1 Introduzione

In questo nuovo capitolo vengono descritti quali sono le possibili piste da seguire utilizzando quelle che sono le basi, descritte nel capitolo precedente. Si parlerà di nuove tecnologie da utilizzare per la raccolta dei dati, di come ampliare il progetto utilizzando nuovi algoritmi, dell'interazione dei gemelli con l'uomo e di tutti i benefici che questi portano sia all'uomo che all'ambiente

### 3.2 Evoluzione del progetto

Come abbiamo detto nell'introduzione al caso studio del precedente capitolo il progetto che è stato sviluppato costituisce la base di quella che potrebbe essere una espansione molto più ampia e sofisticata. Ad esempio come prima cosa è possibile realizzare un modello 3D della stanza che viene posta in esame, la quale ci servirà per il controllo da remoto e costituirà la parte più concreta del gemello digitale; poi è possibile pensare all'impiego di molti più sensori rispetto a quelli utilizzati nella esercitazione precedente, sarebbe potrebbero utilizzare sensori che rilevano la presenza, che rilevino l'intensità ed il colore della luce, sensori che riescano a riconoscere le persone tramite l'utilizzo della tecnologia face recognition e riescano ad individuare la loro posizione all'interno della stanza, sensori per la rilevazione dei suoni e delle voci e tantissimi altri. Tutti questi sensori rilevano enormi quantità di dati in maniera continua, e quindi si necessita di algoritmi e gemelli digitali costruiti ad hoc per l'elaborazione di tutte queste informazioni, in maniera tale da ottenere il risultato desiderato cioè quello di migliorare la qualità ed il confort della stanza, attraverso delle analisi predittive. Quindi come è possibile immaginare uno degli obiettivi è quello di rendere il gemello digitale completamente autonomo e capace di capire chi è entrato nella stanza, conoscere le sue abitudini ed analizzare la situazione, capire in base alla sua espressione facciale ed ai suoi movimenti e posizione nello spazio, il suo stato d'animo; quindi impostare la stanza cioè accendere luci di varie tonalità di colore, attivare la musica preferita, accendere il televisore, in maniera tale che la persona si rilassi se arrabbiato, o che mantenga il suo stato d'animo felice se il soggetto si trova in un momento di felicità. Inoltre è possibile anche far interagire l'uomo con il sistema, utilizzando dei comandi vocali.

### 3.3 Modello 3D

Come già detto uno dei primi passaggi è quello della trasformazione da reale a virtuale dello spazio di cui si andrà a realizzare il gemello virtuale. Si cerca di realizzare nella maniera più accurata possibile tramite l'impiego di software di modellazione 3D, questa riproduzione perché sarà poi lei, l'interfaccia da cui è possibile monitorare il locale di cui abbiamo realizzato il gemello. Risulta essere di fondamentale importanza, verificare le tecnologie utilizzate per la realizzazione del modello tridimensionale, che esse siano compatibili con quelle che si andranno ad utilizzare per la realizzazione della parte software dei gemelli digitali. La realizzazione del modello è uno dei passaggi più veloci ed immediati di tutte le evoluzioni che abbiamo precedentemente elencato.



**Fig.41**

### 3.4 Sensoristica

Nell'introduzione sono stati citati varie tipologie di sensori dai più banali fino a sensori davvero complessi, prima di tutto diamo una definizione di quello che sono i sensori, “sono dei dispositivi in grado di rilevare una grandezza interagendo con essa: l'energia ricevuta dal sensore modifica lo stato della grandezza variando una delle sue proprietà. Dato che all'interazione con la grandezza è associata in genere anche l'operazione di conversione, è possibile definire sensore un dispositivo che acquisisce in ingresso una certa grandezza e ne fornisce in uscita una differente, ma legata alla prima da una legge nota” [2]

L'interesse per il campo della sensoristica è sempre in continua crescita, ciò è dovuto al fatto che si ha una esigenza di misurare grandezze fisiche, chimiche e biologiche in numero molto ampio di aree applicative: dal monitoraggio degli strumenti, analisi e diagnostica, ambito medico ecc... [2]

È possibile fare una suddivisione tra tutte le tipologie di sensori fisici: Meccanici, Termici, Acustici, Elettrici, Magnetici, Ottici, Elettromagnetici. Ognuno di essi rileverà delle grandezze differenti

Essi possono misurare e mostrare direttamente la grandezza fisica in esame, essere collegati ad uno strumento che ci mostra il valore misurato (display), oppure possono essere collegati ad uno strumento che registra i dati, come nel caso in esame, avviene una trasformazione analogico-digitale del segnale e poi memorizzato in una macchina ad esso collegata. Questo processo di conversione da analogico a digitale, esercita due operazioni di discretizzazione sulla grandezza, una di **campionamento** e l'altra di **quantizzazione**. Se questa operazione viene guardata dal punto di vista elettronico, i principali attori sono un convertitore A/D il quale è gestito da un microprocessore, il quale può essere una scheda di acquisizione completamente gestita dal computer, per ricevere il segnale questa porta sarà dotata di una porta di ingresso analogica, attraverso la quale confluiscono i segnali in uscita dai sensori. Il problema dell'acquisizione può essere accentuato qualora si hanno in ingresso più sensori che forniscono informazioni simultaneamente, come nel caso di cui si è parlato al paragrafo [3.2]. In queste situazioni è necessario dotarsi di una scheda multicanale per l'acquisizione dei dati, il sistema host memorizzerà i dati su delle tabelle diverse in base alla tipologia del dato, in maniera tale che quest'ultimi possano essere sempre interpretati nella maniera corretta ed inoltre i valori simultanei delle diverse grandezze possano essere sempre riconosciuti.



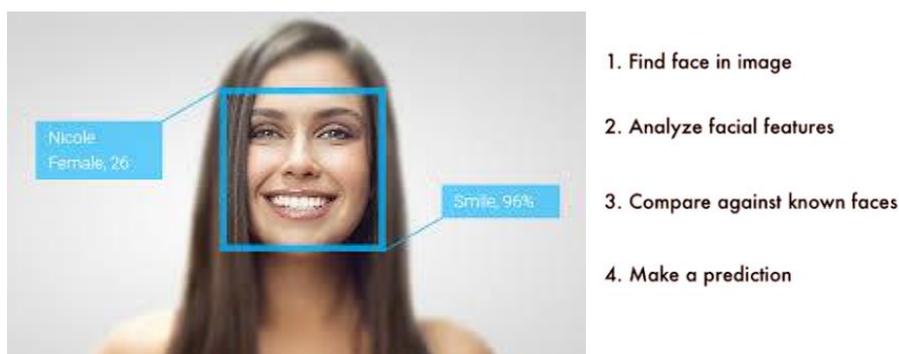
**Fig.42**

### 3.4.1 Face Recognition e telecamere

Nei precedenti capitoli sono stati elencati diverse tipologie di sensori, ma quello che sicuramente avrà suscitato maggiore interesse e stupore, è il sensore che ci consente di eseguire un riconoscimento facciale e quindi di poter permettere al sistema di capire con chi si deve relazionare. Ora si cercherà di capire come questa nuova tecnologia funzioni e quali sono le sue applicazioni. Il face recognition consiste in una serie di problemi diversi ma che sono correlati tra loro, essi possono essere elencati:

- Trovare le facce all'interno di una foto.
- La seconda fase consiste nel concentrarsi su una faccia ed essere in grado di capire che si tratta sempre della stessa persona, anche se essa si trova in una posizione strana o magari il suo viso non è illuminato nella maniera corretta.
- Riuscire ad individuare dei punti, che risultino essere delle unicità del viso, in maniera tale che possano essere utilizzati per distinguere il soggetto da altre persone, un esempio può essere la larghezza del viso, oppure la dimensione delle labbra o degli occhi ecc.
- Infine, debbono essere confrontate tutte le caratteristiche di cui abbiamo parlato sopra, con i volti e le persone di cui già si è a conoscenza per andare a determinare il nome del soggetto individuato.

Diciamo che il nostro cervello elabora tutte queste informazioni in un tempo brevissimo e in maniera automatica, tanto che a volte l'uomo vede volti raffigurati in oggetti oppure riconosce dei volti perché magari riconosce dei segni particolari di una determinata persona sul viso di altre persone. Oggi però i computer non sono così intelligenti da percorrere tutto questo percorso logico in maniera autonoma, l'uomo tramite l'utilizzo di algoritmi, insegna alla macchina come riconoscere un volto. Questo insegnamento avviene risolvendo passaggio per passaggio tutte le problematiche sopra elencate, ed i risultati del primo passeranno al secondo e così via fino a raggiungere l'obiettivo.

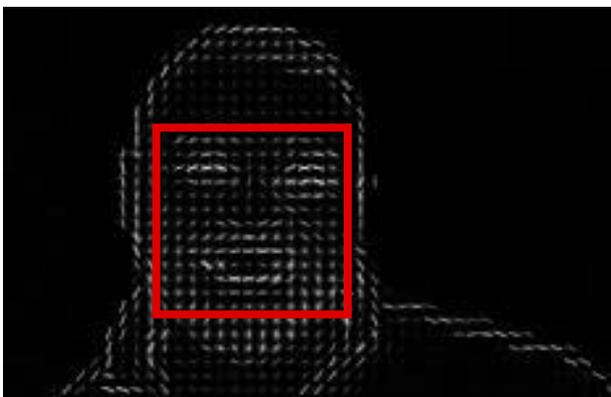


**Fig.43**

### 3.4.1.1 Trovare i volti nell'immagine

Quindi come abbiamo già detto il primo passaggio è quello di riuscire ad individuare e localizzare tutti i volti che sono presenti nell'immagine che si sta analizzando. Ci sono molti metodi per il rilevamento dei volti, i primi sono stati ideati nei primi anni 2000 gli ideatori furono Paul Viola e Micheal Jones, poi con il passare del tempo e con l'avanzare delle tecnologie e la riduzione dei costi si è potuto sviluppare nuovi metodi.

Per iniziare con il processo di localizzazione del volto si parte con il rendere l'immagine in bianco e nero, perché non sono necessari i colori in questa fase, quindi si andranno ad esaminare ogni singolo pixel della foto e tutti quelli che lo circondano, per capire l'intensità del colore e verso che direzione il l'immagine diventa più scura. Una volta compresa la direzione si applica una freccia che viene chiamata **gradiente**. Potrebbe sembrare una cosa inutile e noiosa ma invece si rivela di fondamentale importanza, perché una volta che sono stati individuati tutti i gradienti, se dovesse capitare una foto più chiara o più scura le direzioni dei gradienti non variano. Per semplificare il tutto si adottano dei quadrati 16x16 pixel e in base a quali sono i gradienti dominanti, si assegnerà la freccia dell'insieme. L'immagine che otterremo sarà la seguente e questo tipo di rappresentazione è chiamata HOG (Histogram of Oriented Gradients).



**Fig.44**

### 3.4.1.2 Analisi delle caratteristiche facciali

Una volta che si è riusciti ad isolare i volti nell'immagine, si deve affrontare il problema dell'orientamento del volto, quando stessi soggetti raffigurati in immagini differenti sono rivolti in maniera diversa, per il computer sono due persone differenti, quindi è nostro compito allenare la macchina e non commettere questo errore.

Per poter eseguire questo passaggio, si procede con il deformare qualsiasi immagine in maniera tale che gli occhi e le labbra si trovino sempre nella stessa posizione campione. Di seguito verrà utilizzato un algoritmo denominato **stima del punto di riferimento del viso** inventato da Vahid Kazemi e Josephine Sullivan nel 2014. Non è l'unico metodo per fare questo tipo di lavoro ne esistono molti altri.

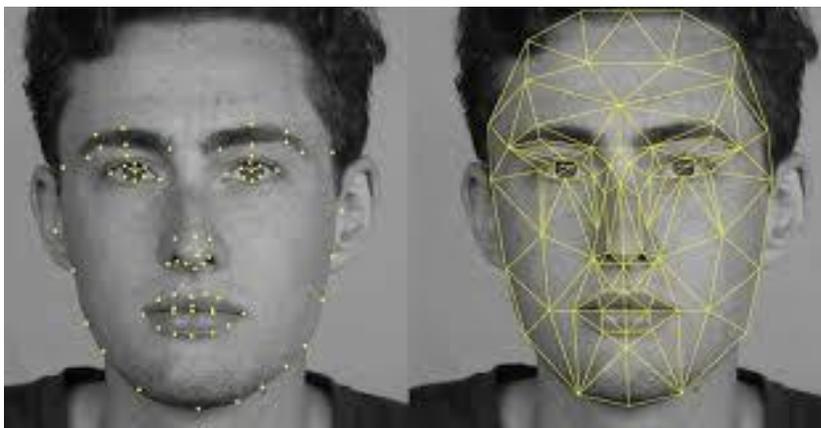
L'algoritmo ideato funziona sulla base di 68 punti specifici che comunemente vengono chiamati **punti di riferimento** i quali sono presenti su ogni volto, dalla parte superiore del mento, al contorno delle sopracciglia, profilo del naso ecc. La macchina una volta individuati questi 68 punti di riferimento si addestrerà a trovarli, quindi anche se il soggetto analizzato avrà un orientamento del viso diverso ogni volta, la macchina sarà in grado di centrare quei dettagli che sono stati individuati.



Fig.45

### 3.4.1.3 Codifica dei volti

Questa è una delle fasi più delicata delle quattro, perché è necessario risolvere il problema della distinzione dei volti. Il metodo più semplice è quello di confrontare tutti i dettagli che abbiamo individuato nella precedente fase, con le immagini delle persone che sono state già etichettate; quando si trova un volto simile allora è per forza la persona a noi sconosciuta. In realtà questo metodo essendo il più facile a livello concettuale è anche il più difficile e sconveniente da utilizzare quando si adopera per un elevato numero di persone, perché se si dovesse confrontare miliardi di immagini, si impiegherebbe troppo tempo. Un altro tipo di approccio potrebbe essere quello delle misurazioni, in questa maniera si determinano quali sono le dimensioni dell'orecchio oppure quelle della bocca ecc. In realtà si è scoperto che a scegliere quali parti misurare non è compito umano, perché potrebbe sbagliare, ma bensì è il computer che in maniera autonoma, tramite approfondimento profondo, determina quali sono le parti migliori per effettuare le misurazioni. L'algoritmo per le misurazioni dovrà essere allenato a fare 128 misurazioni per ogni volto, quindi il suo allenamento consisterà nell'effettuare queste misurazioni in un volto conosciuto, poi altre 128 misurazioni in un volto sempre conosciuto ma in posizione differente, poi ultimo passaggio, gli viene data in pasto una immagine di un volto che non conosce e anche su questo deve effettuare 128 misurazioni, una volta effettuate queste misurazioni deve capire che la prima e la seconda foto hanno delle misure che si assomigliano mentre la terza ha dei valori più distanti rispetto ai precedenti. Tutto questo deve essere ripetuto per milioni di immagini e persone in maniera tale che la rete sia ben allenata nella generazione di 128 misurazioni e che anche se si prendo una dozzina di immagini diverse della stessa persona le misurazioni effettuate sulle immagini risultino molto molto simili. L'addestramento della rete è un processo abbastanza lungo, se disponiamo di un'ottima scheda video il processo potrebbe durare 24 ore consecutive. Fortunatamente è possibile scaricare delle reti già addestrate le quali possono essere utilizzate direttamente.

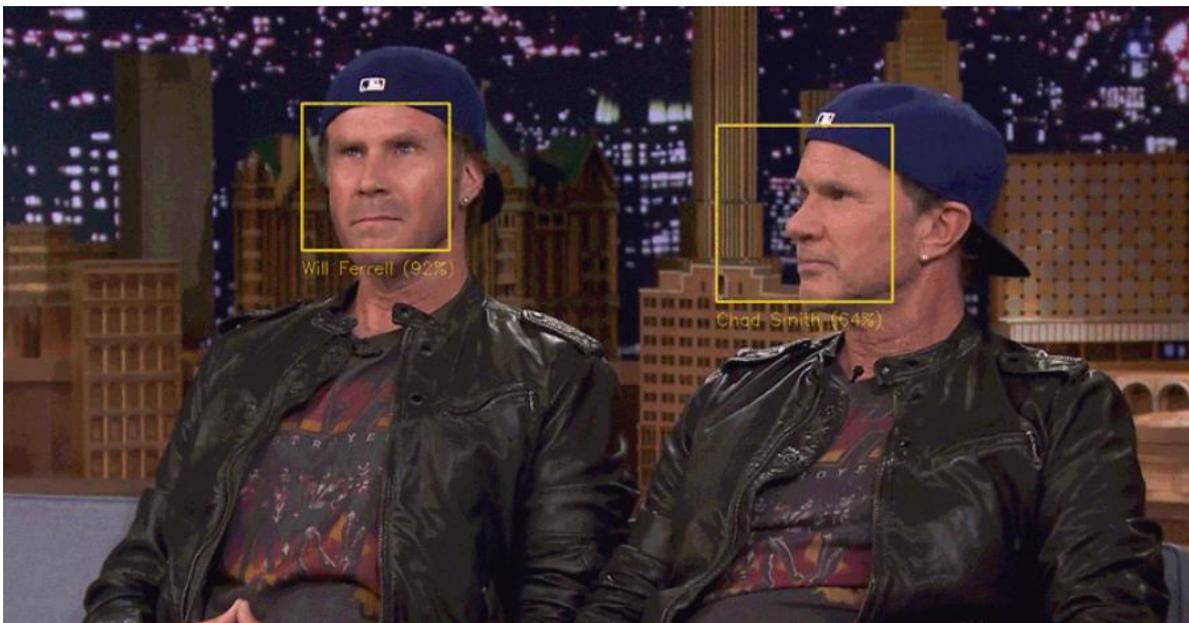


**Fig.46**

#### 3.4.1.4 Identificare la persona raffigurata nell'immagine

Questo passaggio è il più semplice ed immediato dei quattro in esame, consiste infatti nell'andare a cercare e trovare l'identità della persona analizzata, che sarà presente in un database di persone note, le quali hanno le misure del viso più simili a quelle rilevate.

Per poterlo mettere in atto questo passaggio, è necessario adottare un qualsiasi algoritmo di machine learning di database. Tutto quello che è necessario fare, consiste nel prendere un classificatore, il quale prenderà delle misurazioni da un volto nuovo raffigurato in una nuova immagine, esso dovrà riuscire a decretare a quale persona conosciuta si avvicina di più.



**Fig.47**

#### 3.4.1.5 Avvertenze

I modelli di riconoscimento facciale potrebbero non essere adatti, per l'impiego con i bambini, questi tendono a confondersi quando incontrano visi le cui distanze potrebbero cambiare rapidamente nel tempo.

Inoltre la sua precisione può dipendere anche da gruppo etnico che lo adopera, come è stato detto in precedenza il modello è tanto più preciso quanti più dati elabora ma dato che esso si allena utilizzando dei set di dati pubblici, con immagini prese da siti web, queste immagini potrebbero non essere distribuite in maniera uniforme tra tutti gli individui di tutti i paesi, quindi è probabile che vengano riconosciuti meglio dei lineamenti occidentali piuttosto che quelli asiatici o viceversa.

### 3.4.2 Human Activity Recognition (HAR)

Il principale obiettivo di questi sistemi HAR è quello di osservare e studiare tutte quelle che sono le attività umane, interpretando tutti gli eventi che accadono in quello spazio correttamente. Molto spesso fanno uso di sensori visivi, ma possono anche non essere utilizzati, per la raccolta ed elaborazione delle informazioni ambientali, temporali, spaziali ecc. per riuscire a capire il comportamento dell'uomo in esame. Questi sistemi vengono adottati qualora si intenda andare ad analizzare tutte quelle informazioni necessarie allo studio del comportamento umano in quel determinato ambiente (quindi giocherebbe un ruolo fondamentale nel caso studio del gemello digitale della stanza). Le informazioni che vengono estratte possono essere diverse, ora ne elenco qualcuna:

- **Presenza**, ci è possibile verificare se all'interno della stanza vi è traccia di un individuo (non è da escludere la questione sicurezza).
- **Numero**, il sistema riesce a contare il numero di persone presenti all'interno dello spazio che è sottoposto a verifica, basti pensare al periodo che stiamo attraversando in cui non bisogna creare sovraffollamento nei locali chiusi, questa tecnologia sarebbe davvero di aiuto per chi si occupa della gestione dei locali.
- **Tracking**, è possibile tracciare quali spostamenti vengono effettuati dalla persona inquadrata e non solo, è possibile analizzare anche la postura del soggetto, il tutto al variare del tempo.
- **Identità**, informazione di cui abbiamo abbondantemente parlato nel capitolo precedente.

Il principale campo di applicazione di questa tecnologia è l'Ambient Assisted Living (AAL), il quale rappresenta un modo per descrivere quell'insieme delle tecnologie non intrusive atte a rendere l'ambiente nel quale viviamo, collaborativo ed intelligente, il tutto finalizzato sempre al miglioramento della qualità della vita per le persone.

Adottando queste tecnologie è possibile avere dei benefici come ad esempio aumentare il periodo in cui le persone possono utilizzare l'ambiente, aiutare le persone anziane a compiere gesti o azioni per loro difficili e pericolose, migliorare lo stile di vita, garantire sempre un collegamento con ad esempio i soccorsi in caso di incidenti.

In questo settore è molto importante il concetto di Activity of Daily Life (ADL), in questo concetto sono racchiuse tutte quelle azioni che comunemente vengono svolte all'interno di una abitazione e ci consente di costruire un sistema capace di prevedere quello che la persona andrà

a fare all'interno di questo ambiente e quindi possiamo costruire degli algoritmi che ci consentano di rendere il servizio di assistenza automatico.



Fig.48

## 4. Conclusioni

In conclusione lo studio ed il lavoro svolto in questa tesi erano finalizzati alla realizzazione di un progetto che funga da base per la realizzazione del Digital Twin in un ambiente domestico. In particolare mi ha portato a studiare ed analizzare lo stato dell'arte del mondo dei Digital Twin e dell'IoT, e le tecnologie che permettono lo sviluppo e la realizzazione di questi Gemelli. Grazie a questi studi che sono stati condotti sulle tecnologie da applicare nel campo dei Gemelli Digitali sono riuscito ad apprendere ed addestrarmi nell'utilizzo della piattaforma Azure con la quale è possibile realizzare quanto fatto e non solo.

In seguito ai risultati ottenuti dalla realizzazione del progetto è possibile affermare che il sistema simulato comprensivo di tutti i dispositivi necessari funziona nella maniera corretta, le informazioni di temperatura che vengono rilevate dal termostato vengono successivamente inviate tramite l'applicazione all'hub il quale tramite il grafo invia questi dati ai gemelli, il tutto si svolge come da programma e le informazioni che vengo stampate alla fine dal gemello digitale sono estremamente corrette.

Come detto più volte la simulazione comprende un solo sensore e quindi i dati elaborati dal gemello risultano essere solo quelli della temperatura, qualora si decida di inserire altri sensori è necessario creare un nuovo schema ed effettuare tutti i collegamenti necessari in maniera tale che i gemelli possano elaborare tutti i dati.

Per quanto riguarda Microsoft Azure, è possibile dire che sia una eccellente piattaforma sia dal punto di vista della semplicità di utilizzo, perché molto intuitiva, sia per una questione di quantità di informazioni e documenti messi a disposizione dalla stessa Microsoft per spiegare come approcciarsi all'utilizzo, le sue guide contengono esercitazioni molto formative ed utili. Nel giro di poco tempo è possibile anche per le persone con poca esperienza nell'ambito della programmazione e dell'informatica, riuscire a muoversi con agilità all'interno della piattaforma ed eseguire tutte le varie operazioni, seguendo le guide. Inoltre è possibile anche fare una valutazione economica della piattaforma, in circa due mesi e con tutte le risorse utilizzate la spesa si aggira intorno a 20\$ dei 100\$ che avevamo a disposizione con la nostra licenza.

Analizzando anche quello che sono le possibili evoluzioni future è possibile affermare che sicuramente aumenteranno la complessità del sistema, ma sicuramente renderebbero il progetto molto innovativo ed anche possibile da attuare fisicamente.

## Sitografia e referenze

<https://newsimpresa.it/value4manager/2018/04/digital-twin-la-storia-un-gemellaggio-digitale/>

<https://www.researchgate.net/publication/307509727> Origins of the Digital Twin Concept

<https://www.researchgate.net/publication/336870688> Enabling technologies and tools for digital twin

<https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cfc121d78>

[https://www.treccani.it/enciclopedia/sensori\\_%28Enciclopedia-della-Scienza-e-della-Tecnica%29/](https://www.treccani.it/enciclopedia/sensori_%28Enciclopedia-della-Scienza-e-della-Tecnica%29/) [2]

<https://docs.microsoft.com/en-us/azure/digital-twins/tutorial-code>

<https://www.researchgate.net/publication/333573394> Re-design of smart homes with digital twins

<https://adalidda.com/posts/wR2E8TgzF2dsSK4F3/re-design-of-smart-homes-with-digital-twins>

[https://books.google.it/books?hl=it&lr=&id=WfOFDwAAQBAJ&oi=fnd&pg=PT3&dq=benefici+attesi+digital+twin&ots=EciQIJ\\_I9F&sig=46mBAN\\_u0ZS7LgiYNV5EuUhoLhE#v=onepage&q&f=false](https://books.google.it/books?hl=it&lr=&id=WfOFDwAAQBAJ&oi=fnd&pg=PT3&dq=benefici+attesi+digital+twin&ots=EciQIJ_I9F&sig=46mBAN_u0ZS7LgiYNV5EuUhoLhE#v=onepage&q&f=false)

[https://amslaurea.unibo.it/12946/1/Tesi\\_Andrea\\_Zagnoli\\_HumanActivityRecognition.pdf](https://amslaurea.unibo.it/12946/1/Tesi_Andrea_Zagnoli_HumanActivityRecognition.pdf)

libro PLM Product Lifecycle Management: Driving the Next Generation of Lean Thinking [1]

## Ringraziamenti

Giunti alla conclusione di questa tesi e di questo ciclo è giusto ringraziare tutte le persone che mi sono state vicino in qualsiasi situazione bella o brutta che sia. Per primi vorrei ringraziare i miei genitori, mio padre Luciano e mia madre Monica e mio fratello Luca grazie ai quali ho potuto iniziare questo cammino e durante tutto il percorso non mi hanno mai fatto sentire solo, ci sono sempre stati ed hanno camminato insieme a me. Poi vorrei ringraziare tutto il resto della famiglia dagli zii, i cugini e le nonne che grazie ai loro pranzi e cene mi hanno fatto distrarre un po' da quelle infinite domeniche o giorni di festa passati a studiare. Ormai fa parte della famiglia anche lei ma merita un ringraziamento a parte, e questo è per Chiara la mia ragazza, con la quale ho condiviso due terzi di questo cammino ma che anche lei mi è sempre stata vicino, con le sue parole mi dava la carica e mi trasmetteva calma e serenità quando queste a volte venivano meno, non solo le parole ma anche fisicamente mi era vicino in quei periodi in cui dovevo rimanere per qualche settimana a Fermo, lei faceva il suo bagaglio e mi veniva a trovare, per dimostrarmi che lei era con me pronta al sostegno. Ringrazio anche tutta la sua famiglia, che mi sempre mostrato affetto e conforto. Poi ci sono loro, quelli che non sono amici solo il sabato ma ogni giorno, quelli che se hai un problema si fanno in quattro per aiutarti, quelli che anche se il sabato dicevo no alla cena o all'aperitivo perché dovevo studiare mi hanno sempre capito e sostenuto, quindi ringrazio: Francesco (Zampa), Lorenzo (Format), Mirco (Moscio), Ledjan (Letti), Erion (Eri), Filippo (Renz), Lisandro (Lis), Nicola (Sintom), Nicola (Tetta), Matteo (Mendi), Stefano (Stè), Samuele (SamuACC), Giorgia (Giò), Alice (Ali), Laura (Lalla), Arianna (Bruna), Maria (Mari). Ora è il momento di ringraziare coloro i quali hanno condiviso con me pranzi e cene (Lunedì: capocollo e patate), gioie infinite e litigate impazzite, chi si è sopportato le mie sveglie alle 5 e i vari Guccini, Vecchioni e De André, le urla e le mie fisse, grazie DIFFEOMORFISMO, Seba, Sibi, Anna sarete sempre nel mio pensiero ogni lunedì quando mangerò le patate e nessuno me le ruberà (a breve vi svelerò la ricetta). Ora è il turno di ringraziare due persone molto speciali, che se sono arrivato dove sono è anche grazie a loro, che hanno saputo gestire la mia testardaggine in maniera eccellente, aiutare quando i conti non portavano e passare gli appunti quando c'erano dei buchi nei miei, quante arrabbiate ma anche quanti sorrisi fatti, quante emozioni, quante parole di sostegno e quante chiamate per spiegarsi gli esercizi o per cambiare le cose nei lavori di gruppo ad orari improponibili. Vi voglio bene amiche mie Elena (Elenona), Lucia (LuciPazz). Non manco nel ringraziare tutti gli amici di Fermo e Jacopo con il quale abbiamo macinato chilometri su chilometri per raggiungere la città. Ultimi ma non per importanza anzi, vorrei dedicare un pensiero speciale ai miei nonni Mario e

Nino, i quali purtroppo non possono essere presenti per questo passo della mia vita ma che sicuramente stanno gioendo per il traguardo che sto per raggiungere, anche perché quando ero piccolo entrambi mi dicevano di intraprendere ingegneria.

Poi vorrei ringraziare la Prof.essa Maura Mengoni e il Dott. Alex Altieri per il sostegno, la disponibilità, e la professionalità mostrati nei miei confronti durante il tirocinio e tesi.

Concludo ringraziando un professore delle medie che con le sue parole mi spinse a dare ogni giorno il massimo per raggiungere i miei obiettivi e dimostrare che si giudica sempre alla fine dei lavori, riporto la sua frase che per me è stata di una carica infinita “Se non ti bocchiamo quest’anno, ricordati che accadrà il prossimo anno”

Per tutti coloro che leggeranno questa tesi e non sono stati citati non me ne vogliate ma sappiate che ringrazio anche voi.

**GRAZIE A TUTTI.**