

RELAZIONE

Componenti del gruppo:

1. Componente 1: *Pietro Mezzatesta 20054585*
2. Componente 2: *Ossama Boulakhdar 20054129*

Progettazione Concettuale:

1.1. Requisiti iniziali

Requisiti iniziali: Piattaforma per la Human-Robot Interaction

La Human-Computer Interaction (HCI) è una branca della computer science che si occupa del design, dello sviluppo e dello studio delle **interazioni tra esseri umani e sistemi tecnologici**, con l'obiettivo di rendere queste interazioni **intuitive, efficienti e soddisfacenti**. Nata negli anni '80 con la diffusione dei personal computer, l'HCI ha ampliato il proprio campo di studio per includere aree come **l'intelligenza artificiale**, la realtà aumentata, la robotica e le tecnologie per il **dialogo naturale**. Un aspetto cruciale dell'HCI contemporanea è il **dialogo emotivo tra umano e robot**, che mira a creare interazioni non solo funzionali, ma anche **emotivamente coinvolgenti**. Questo approccio sfrutta **tecniche di riconoscimento e generazione di emozioni per costruire interfacce capaci di adattarsi allo stato d'animo dell'utente**. Ad esempio, un sistema che percepisce frustrazione o ansia in un utente può rispondere con toni empatici e rassicuranti, migliorando l'esperienza complessiva. In questo contesto, l'uso della **persuasione** rappresenta un elemento innovativo e potente. La persuasione nell'HCI riguarda **l'adozione di strategie per influenzare le decisioni e i comportamenti degli utenti in modo positivo, etico e mirato**. Nel dialogo emotivo, un sistema persuasivo non si limita a fornire informazioni, ma **cerca di guidare l'utente** verso scelte benefiche, come adottare abitudini più salutari, completare un compito complesso o migliorare il proprio stato d'animo. Un esempio pratico può essere un assistente virtuale che incoraggia l'utente a prendersi una pausa durante una giornata stressante, utilizzando un **linguaggio** motivante ed empatico. La persuasione efficace, **si basa su una comprensione approfondita delle emozioni** (emotional alignment), della **psicologia dell'utente** e del contesto dell'**interazione**. Strumenti come il riconoscimento vocale e facciale, combinati con modelli di machine learning, consentono ai sistemi di personalizzare i messaggi persuasivi in tempo reale (tecnologie per interfaccia canale). Ad esempio, un chatbot progettato per promuovere uno stile di vita sano può adattare i

propri suggerimenti in base all'umore e alle abitudini dell'utente, creando un'esperienza più coinvolgente e personalizzata. L'integrazione della persuasione nelle interazioni uomo-macchina solleva, tuttavia, questioni etiche. È fondamentale che queste tecniche vengano utilizzate in modo trasparente e rispettoso, evitando manipolazioni indebite. L'HCI si impegna a garantire che i sistemi persuasivi siano progettati con un approccio user centered, ponendo al centro i bisogni e i diritti dell'utente. In sintesi, l'HCI, combinata con tecniche persuasive nel dialogo emotivo, rappresenta una frontiera entusiasmante per creare sistemi tecnologici capaci di comunicare in modo umano, empatico e costruttivo, migliorando non solo l'efficacia delle interazioni, ma anche il benessere degli utenti. Negli ultimi anni, l'evoluzione tecnologica, gli LLMs (Large Language Models) come i sistemi di OpenAI ChatGPT e AI generativa, hanno portato a un'interazione sempre più stretta tra esseri umani e sistemi artificiali. I robot, inizialmente concepiti come macchine per eseguire compiti ripetitivi o pericolosi, stanno rapidamente diventando compagni intelligenti capaci di interagire con noi in modo sofisticato. Questa trasformazione è resa possibile grazie ai progressi nell'intelligenza artificiale, nell'apprendimento automatico e nell'elaborazione del linguaggio naturale. Un aspetto cruciale per rendere queste interazioni più umane è la capacità dei robot di riconoscere, interpretare e rispondere alle emozioni umane. Le emozioni giocano un ruolo fondamentale nella comunicazione umana, influenzando non solo ciò che diciamo, ma anche come lo diciamo. Un sistema in grado di gestire il dialogo emotivo può migliorare significativamente l'esperienza utente, rendendo le interazioni più naturali, empatiche e personalizzate. La progettazione di un database che supporti queste funzionalità è un'impresa complessa che richiede una chiara definizione dei requisiti iniziali. Il database non è semplicemente un archivio di informazioni, ma un componente critico che deve gestire dati strutturati e non strutturati, supportare processi in tempo reale e garantire sicurezza e scalabilità. Questo progetto mira a definire dettagliatamente i requisiti funzionali e non funzionali necessari per la progettazione di un database relazionale, ponendo le basi per lo sviluppo di sistemi robotici avanzati in grado di interagire emotivamente con gli esseri umani.

- Sintesi Funzionale:

Si vuole costruire una base di dati per gestire le interazioni (entità) tra esseri umani e sistemi intelligenti (entità), come macchine dotate di intelligenza artificiale. Tali interazioni devono essere il più dinamiche e naturali possibile ma soprattutto emotivamente coinvolgenti, in quanto il fattore emotivo nelle interazioni umane è fondamentale. Per far avvenire ciò si sfruttano tecniche di riconoscimento e generazione di emozioni (entità), per adattare le interfacce allo stato d'animo

dell'utente (entità). Si adottano diverse tecnologie (il riconoscimento vocale e facciale, modelli di machine learning) per le **interfacce canale** (entità).

La **persuasione** ha un ruolo essenziale nelle interazioni, essa è la capacità di ottenere la fiducia e l'approvazione di un individuo attraverso un'opera di convinzione, in modo da fargli compiere azioni positive, etiche e mirate. Quindi la base di dati deve poter gestire tali interazioni uomo-macchina, **rilevando e comprendendo le emozioni dell'utente in maniera approfondita e rispondendo di conseguenza**, adattandosi al suo stato d'animo. Dunque si avranno le entità: **utente, interazione, robot e emozione**. Durante l'interazione tra umano e robot si dovrà anche rilevare l'intensità dell'emozione, in modo da regolare adeguatamente la **risposta del robot** (entità) per rendere l'esperienza dell'utente più **emotivamente coinvolgente**. **I suggerimenti** (entità) dati dal robot devono essere etici e coerenti con l'umore e la **personalizzazione del robot** (entità) dell'utente. La **sicurezza** è altrettanto importante, in quanto queste interazioni sollevano questioni etiche, per evitare manipolazioni inopportune. Questi sistemi intelligenti devono essere progettati con un approccio user-centered!

Le ultime tecnologie, come LLMs consentano che le interazioni tra persone e macchine siano sempre più **naturali**. I robot stanno diventando parte integrante della nostra vita e lo saranno sempre di più.

L'intelligenza artificiale ha un ruolo centrale, la **capacità dei robot di riconoscere le emozioni e il contesto culturale dell'utente** è la colonna portante per rendere queste interazioni più naturali possibile.

1.2. Glossario dei termini

TERMINE	DESCRIZIONE	SINONIMI	COLLEGAMENTI
Human-Computer Interaction	Branca della computer science che studia e sviluppa le interazioni tra esseri umani e sistemi tecnologici, con l'obiettivo di renderle efficienti e intuitive.	Interazione uomo-computer / persona-computer.	Dialogo emotivo, riconoscimento delle emozioni.
Human-Robot Interaction	Sottocampo dell'HCI focalizzato specificamente sullo studio e lo sviluppo delle interazioni tra esseri	Interazione uomo-robot.	Dialogo emotivo, riconoscimento delle emozioni.

	umani e robot.		
Utente	Colui che interagisce col sistema robotico.	Umano, user.	Robot, interazione, Sessione
Robot	Dispositivo dotato di capacità interattive e emotive.	Computer, sistema tecnologico/robotico, I.A., sistema, macchina.	Utente, interazione, Sessione
Interazione	Scambio comunicativo tra utente e robot.	Dialogo, messaggio, comunicazione.	Utente, robot, Sessione
Emozione	Stato emotivo percepito/rilevato.	Sentimenti, stato d'animo.	Interazione, emotional alignment
Persuasione	Strategia utilizzata dal sistema per influenzare positivamente le decisioni degli utenti.	Influenza positiva, guida decisionale.	Interazione
Emotional alignment	Processo di comprensione e allineamento con le emozioni dell'utente per personalizzare l'interazione.	Riconoscimento delle emozioni.	Emozione, riconoscimento delle emozioni
Riconoscimento delle emozioni	Capacità di un sistema di comprendere lo stato emotivo dell'utente attraverso vari input.	Analisi emotiva.	Emotional alignment, emozione, messaggio
Dialogo emotivo	Tipo di interazione tra umano e robot che mira a creare comunicazioni non solo funzionali ma anche emotivamente coinvolgenti.	interazione emotiva.	Emotional alignment, riconoscimento delle emozioni.
LLMs (Large Language Models)	Modelli avanzati di intelligenza artificiale, come ChatGPT di OpenAI, capaci di comprendere e generare linguaggio naturale.	Modelli di linguaggio/linguistici avanzati.	AI generativa, interazione uomo-macchina.
AI generativa	Tecnologia di intelligenza artificiale in grado di creare contenuti originali come testo, immagini o audio in risposta agli input dell'utente	IA, Intelligenza artificiale	LLMs

Intelligenza Artificiale	Campo della computer science che si occupa di sviluppare sistemi capaci di eseguire compiti che tipicamente richiedono intelligenza umana.	IA, machine learning	Machine learning, elaborazione del linguaggio naturale.
Machine Learning	Sottocampo dell'intelligenza artificiale che permette ai sistemi di apprendere dai dati e migliorare nel tempo senza essere esplicitamente programmati.	Intelligenza artificiale, IA	Intelligenza artificiale, elaborazione del linguaggio naturale
Elaborazione del linguaggio naturale	Tecnologia che permette ai computer di comprendere, interpretare e rispondere al linguaggio umano.	Natural Language Processing (NLP)	Intelligenza artificiale, LLMs
Sicurezza	Protezione dei dati e delle funzionalità del sistema da accessi non autorizzati e altre minacce.	protezione, privacy	Utente, Robot, Sessione, Log errori
Scalabilità	Capacità del database di gestire un aumento del carico di lavoro senza compromettere le prestazioni mantenendo efficienza e dati coerenti.	flessibilità, estensibilità, adattabilità, modularità.	Utente, interazione, sessione, robot
Feedback	Valutazione esplicita che un utente può dare a una risposta	Commento, valutazione	Utente, interazione
Risposta	Output generato dal sistema in risposta a un input dell'utente. Può essere associata ad un'emozione e tradotta in più lingue.	Riscontro, replica, opinione, suggerimento	Emozione, Robot, Traduzione,
Suggerimento	Output proattivo che il sistema fornisce all'utente in base al suo profilo psicologico, al contesto o all'emozione rilevata.	Risposta, opinione	Emozione, Robot
Traduzione	Versione localizzata di una risposta in una lingua diversa rispetto all'originale.	Versione	Risposta
Interfaccia canale	Canale tecnico tramite cui avviene l'interazione	chatbot, assistente vocale, app...	interazione, utente

Sessione	Intervallo di tempo in cui un utente è autenticato e interagisce col sistema.	Connessione/interazione	Robot, interazione, utente
Configurazione Privacy	Insieme di preferenze esplicitate dall'utente riguardo alla gestione dei dati personali.	Privatezza, preferenze, riservatezza	Utente
Contenuto multimediale	Oggetto multimediale utilizzato o prodotto dall'interazione.	Immagine, audio, video	Interazione
Metrica	Indicatore numerico delle prestazioni del sistema (es. tempo medio di risposta, tasso di errore).	Prestazioni	Interazione
Log errori	Registro degli errori rilevati dal sistema durante il funzionamento.	Registrazioni degli errori	Robot, sessione
Politica di moderazione	Insieme di regole e azioni automatizzate per filtrare contenuti inappropriati o violazioni.	Sicurezza, rispetto, moderazione, etica	Robot, risposta, suggerimento
Accessibilità	Serie di impostazioni in grado di garantire inclusività e di essere utilizzato da tutti gli utenti.	Normativa, impostazioni	Utente

Operazioni da effettuare sui dati:

- *Op1*: Il sistema registra un nuovo utente, con profilo psicologico, preferenze, lingua e configurazioni di privacy/accessibilità. (frequenza alta).
- *Op2*: L'utente interagisce con il robot avviando una sessione, attraverso l'interfaccia canale.(frequenza alta).
- *Op3*: Ogni interazione registra l'emozione rilevata, la sua intensità e riceve eventuale feedback. (frequenza molto alta).
- *Op4*: Il robot genera risposte e suggerimenti basati sullo stato emotivo e sul contesto dell'utente.(frequenza molto alta).
- *Op5*: Le risposte sono collegate a emozioni, possono essere tradotte e adattate alla lingua dell'utente. (frequenza alta).

- *Op6*: Il robot possiede competenze, comprensione di input multimodali e può adattarsi al profilo e allo stile comunicativo dell'utente. (frequenza alta).
 - *Op7*: Il robot è personalizzato dall'utente, sia nella configurazione che nel comportamento. (frequenza media).
 - *Op8*: Il sistema archivia e consente la ricerca delle interazioni e dei dati storici. (frequenza alta per archiviazione, bassa per ricerca esplicita).
 - *Op9*: Il sistema elabora i feedback espliciti e impliciti per apprendere e migliorare l'interazione. (frequenza media).
 - *Op10*: Il sistema applica politiche di moderazione per contenuti inappropriati. (frequenza media).
 - *Op11*: Il sistema gestisce sessioni utente sicure, autenticazione e tracciabilità. (frequenza alta).
 - *Op12*: Il sistema supporta interazioni multilingua, traducendo input e output in tempo reale. (frequenza molto alta).
-

1.3. Requisiti riscritti

~~La Human-Computer Interaction (HCI) è una branca della computer science che si occupa del design, dello sviluppo e dello studio delle interazioni tra esseri umani e sistemi tecnologici, con l'obiettivo di rendere queste interazioni intuitive, efficienti e soddisfacenti.~~

La Human-Computer Interaciton (HCI) è una disciplina che studia le interazioni tra utenti e sistemi tecnologici, con l'obiettivo di renderle intuitive ed efficienti.

~~Nata negli anni '80 con la diffusione dei personal computer, l'HCI ha ampliato il proprio campo di studio per includere aree come l'intelligenza artificiale, la realtà aumentata, la robotica e le tecnologie per il dialogo naturale.~~

L'ambito si è evoluto includendo l'intelligenza artificiale, la robotica e le interfacce dialogiche.

~~Un aspetto cruciale dell'HCI contemporanea è il dialogo emotivo tra umano e robot, che mira a creare interazioni non solo funzionali, ma anche emotivamente coinvolgenti.~~

Il dialogo emotivo tra umano e robot mira a creare interazioni funzionali ed emotivamente coinvolgenti.

~~Questo approccio sfrutta tecniche di riconoscimento e generazione di emozioni per costruire interfacce capaci di adattarsi allo stato d'animo dell'utente.~~

Si usano tecniche per riconoscere e generare emozioni e adattare le interfacce allo stato emotivo dell'utente.

~~Ad esempio, un sistema che percepisce frustrazione o ansia in un utente può rispondere con toni empatici e rassicuranti, migliorando l'esperienza complessiva.~~

Quando rileva emozioni come frustrazione o ansia, deve rispondere con tono rassicurante, migliorando l'esperienza dell'utente.

In questo contesto, l'uso della persuasione rappresenta un elemento innovativo e potente. ~~La persuasione nell'HCI riguarda l'adozione di strategie per influenzare le decisioni e i comportamenti degli utenti in modo positivo, etico e mirato.~~

La persuasione è uno strumento integrato nel sistema per indirizzare l'utente verso scelte benefiche, etiche e mirate.

~~Nel dialogo emotivo, un sistema persuasivo non si limita a fornire informazioni, ma cerca di guidare l'utente verso scelte benefiche, come adottare abitudini più salutari, completare un compito complesso o migliorare il proprio stato d'animo.~~

Il sistema suggerisce azioni benefiche in base all'umore e al contesto dell'utente.

~~Un esempio pratico può essere un assistente virtuale che incoraggia l'utente a prendersi una pausa durante una giornata stressante, utilizzando un linguaggio motivante ed empatico.~~

Il sistema deve incoraggiare comportamenti positivi con risposte empatiche.

~~La persuasione efficace, si basa su una comprensione approfondita delle emozioni (emotional alignment), della psicologia dell'utente e del contesto dell'interazione.~~

La persuasione richiede analisi delle emozioni, del profilo utente e del contesto.

~~Strumenti come il riconoscimento vocale e facciale, combinati con modelli di machine learning, consentono ai sistemi di personalizzare i messaggi persuasivi in tempo reale~~

Si usano il riconoscimento vocale, facciale e machine learning per realizzare interfacce intelligenti (tecnologie per interfaccia canale).

~~Ad esempio, un chatbot progettato per promuovere uno stile di vita sano può adattare i propri suggerimenti in base all'umore e alle abitudini dell'utente, creando un'esperienza più coinvolgente e personalizzata.~~

I suggerimenti devono essere personalizzati in base all'umore e al profilo psicologico dell'utente.

~~L'integrazione della persuasione nelle interazioni uomo-macchina solleva, tuttavia, questioni etiche.~~

L'integrazione della persuasione richiede attenzione a trasparenza e rispetto dell'utente.

~~È fondamentale che queste tecniche vengano utilizzate in modo trasparente e rispettoso, evitando manipolazioni indebite.~~

Le tecniche persuasive devono evitare manipolazioni e garantire il rispetto dell'utente.

~~L'HCI si impegna a garantire che i sistemi persuasivi siano progettati con un approccio user centered, ponendo al centro i bisogni e i diritti dell'utente.~~

Il sistema deve essere progettato con approccio centrato sull'utente.

~~In sintesi, l'HCI, combinata con tecniche persuasive nel dialogo emotivo, rappresenta una frontiera entusiasmante per creare sistemi tecnologici capaci di comunicare in modo umano, empatico e costruttivo, migliorando non solo l'efficacia delle interazioni, ma anche il benessere degli utenti.~~

I sistemi tecnologici devono essere capaci di comunicare in modo umano e costruttivo, in modo da migliorare il benessere dell'utente.

~~Negli ultimi anni, l'evoluzione tecnologica, gli LLMs (Large Language Models) come i sistemi di OpenAI ChatGPT e AI generativa, hanno portato a un'interazione sempre più stretta tra esseri umani e sistemi artificiali.~~

I recenti progressi nei LLMs e nell'IA hanno reso le interazioni uomo-macchina più naturali.

~~I robot, inizialmente concepiti come macchine per eseguire compiti ripetitivi o pericolosi, stanno rapidamente diventando compagni intelligenti capaci di interagire con noi in modo sofisticato.~~

I robot stanno diventando entità intelligenti capaci di interagire in modo naturale.

~~Questa trasformazione è resa possibile grazie ai progressi nell'intelligenza artificiale, nell'apprendimento automatico e nell'elaborazione del linguaggio naturale.~~

Tutto ciò è reso possibile da IA, machine learning e NLP.

Un aspetto cruciale per rendere queste interazioni più umane è la capacità dei robot di riconoscere, interpretare e rispondere alle emozioni umane. ~~Le emozioni giocano un ruolo fondamentale nella comunicazione umana, influenzando non solo ciò che diciamo, ma anche come lo diciamo.~~

Le emozioni hanno un ruolo fondamentale nelle interazioni. Ogni interazione deve essere associata a un'emozione.

~~Un sistema in grado di gestire il dialogo emotivo può migliorare significativamente l'esperienza utente, rendendo le interazioni più naturali, empatiche e personalizzate.~~

Il sistema deve gestire emozioni per migliorare la qualità delle interazioni.

~~La progettazione di un database che supporti queste funzionalità è un'impresa complessa che richiede una chiara definizione dei requisiti iniziali.~~

È necessario un database relazionale che supporti emozioni, risposte, interazioni e personalizzazione.

~~Il database non è semplicemente un archivio di informazioni, ma un componente critico che deve gestire dati strutturati e non strutturati, supportare processi in tempo reale e garantire sicurezza e scalabilità.~~

Il database deve gestire dati eterogenei in tempo reale e garantire sicurezza e scalabilità.

Questo progetto mira a definire dettagliatamente i requisiti funzionali e non funzionali necessari per la progettazione di un database relazionale, ponendo le basi per lo sviluppo di sistemi robotici avanzati in grado di interagire emotivamente con gli esseri umani.

1.4. Requisiti strutturati in gruppi di frasi omogenee

Frasi di carattere generale:

Il sistema deve garantire un'interazione uomo-macchina efficace, intuitiva e accessibile. L'obiettivo è realizzare un'esperienza fluida e soddisfacente, capace di adattarsi dinamicamente al profilo emotivo dell'utente. Tutte le funzionalità devono essere progettate secondo un approccio user-centered, tenendo conto della trasparenza, dell'etica e della protezione dei dati. La piattaforma deve essere scalabile, supportare operazioni in tempo reale, registrare eventi e interazioni, e garantire continuità del servizio in ambienti dinamici e multicanale.

Frase relative all'utente:

L'utente può registrarsi nel sistema, specificando preferenze comunicative, lingua, tono, e configurazioni di privacy e accessibilità. Può modificare in ogni momento le proprie impostazioni e interagire attraverso diversi canali, anche in presenza di bisogni speciali. Durante le interazioni, l'utente può fornire feedback espliciti per migliorare la qualità del servizio. Ogni utente ha una sessione autenticata e protetta, durante la quale tutte le interazioni vengono tracciate.

Frase relative al robot

Il robot rappresenta il soggetto interattivo in grado di comprendere, elaborare e rispondere in modo contestuale agli stimoli dell'utente. Ogni robot dispone di un set di competenze e di una configurazione personalizzata, che può includere stile di risposta, livello di empatia e modalità di adattamento all'utente, su parametri contestuali e su obiettivi positivi predefiniti. Le sue azioni devono essere sempre coerenti con i principi etici, evitando comportamenti arbitrari o manipolativi.

Frase relative alle emozioni:

Il sistema è in grado di rilevare una o più emozioni espresse dall'utente durante l'interazione. Ogni emozione è collegata a un livello di intensità e associata temporalmente a uno scambio comunicativo specifico. Le emozioni vengono archiviate nello storico dell'utente e costituiscono un elemento chiave per la personalizzazione delle risposte e dei suggerimenti. La comprensione emotiva è alla base della persuasione positiva e dell'adattamento del sistema ai bisogni dell'utente.

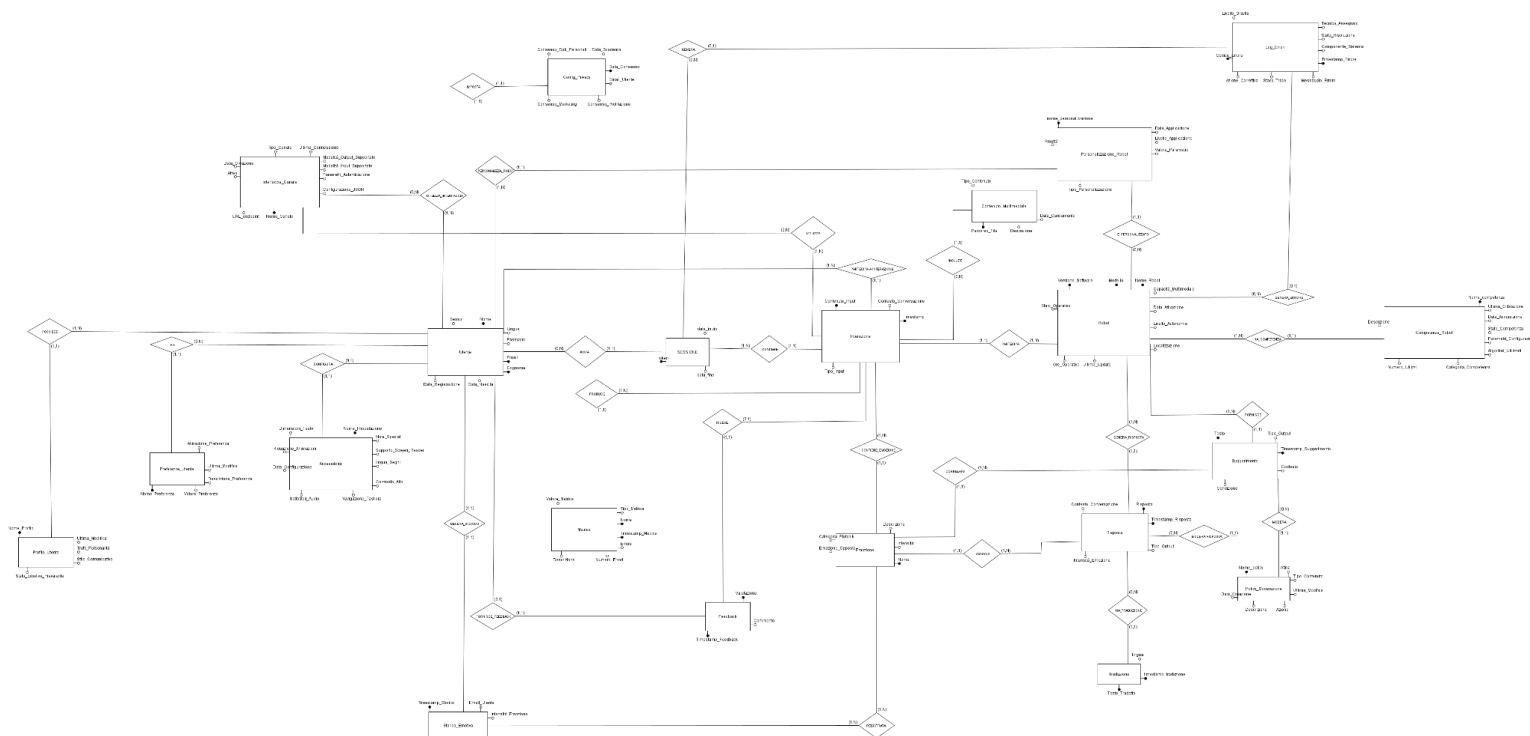
Frase relative all'interazione:

L'interazione tra utente e robot è il nucleo centrale del sistema. Ogni interazione viene tracciata con informazioni relative al canale utilizzato, al contenuto scambiato (anche multimediale), all'emozione rilevata, al feedback fornito e alla risposta generata. Il sistema può operare tramite più canali (app, chatbot, assistenti vocali..) e registra tutti i dati necessari a comprendere il contesto e migliorare la performance futura. Ogni interazione può essere usata per addestrare il sistema, anche in modo implicito, attraverso il monitoraggio delle metriche di risposta e di successo comunicativo.

Frasi relative alla sicurezza e alla moderazione:

Il sistema garantisce la sicurezza delle informazioni sensibili dell'utente, gestisce l'autenticazione e traccia sessioni attive. I dati devono essere trattati secondo le norme vigenti in materia di privacy e sicurezza. Sono presenti meccanismi di moderazione che filtrano contenuti inappropriati o componenti anomali, sia da parte del robot che dell'utente, per assicurare un'interazione eticamente accettabile e priva di rischi. Tutti gli errori e i comportamenti anomali vengono registrati tramite log dedicati.

1.5. Schema E-R + regole aziendali



Commenti schema E-R:

Lo schema E-R progettato rappresenta il modello concettuale di una piattaforma di Human-Robot Interaction focalizzata sul dialogo emotivo e personalizzato tra utente e robot. La struttura del modello riflette i requisiti funzionali e non funzionali specificati, garantendo una rappresentazione completa delle entità coinvolte, delle loro relazioni e dei vincoli semantici.

Lo schema è stato sviluppato secondo un approccio misto, partendo da concetti astratti come “interazione”, “emozione” e “risposta”, per poi derivare entità concrete e relazioni specifiche, realizzando lo scheletro dello schema E-R. In seguito abbiamo decomposto, integrato e raffinato lo schema E-R fino ad arrivare al risultato finale.

Ogni concetto funzionale del sistema è stato rappresentato come entità autonoma o come relazione significativa, rispettando le cardinalità minime e massime indicate dai requisiti.

Analisi dettagliata delle entità:

Utente:

L'entità Utente è il soggetto umano che interagisce col sistema, deve essere identificabile e autenticabile. Esso contiene i dati anagrafici, credenziali, email univoca e lingua preferita. Rappresenta il punto di partenza per tutte le interazioni, consentendo personalizzazione e monitoraggio.

Un utente può avere uno o più profili psicologici, può avere nessuna o più preferenze utente e può decidere di configurare l'accessibilità.

Inoltre un utente al primo accesso è obbligato ad impostare la configurazione privacy un'unica volta, accettando i termini e le condizioni. Un utente può anche utilizzare una o più interfacce canale (ad esempio chatbot su smartphone, interfaccia web, assistente vocale, app mobile...ecc.) in quanto un utente può preferire un canale diverso a seconda della situazione (a casa/in ufficio/in movimento ecc...), avere esigenze diverse (alcuni preferiscono testo, altri voce o video ecc...), e il sistema, come descritto dai requisiti funzionali deve supportare la personalizzazione delle interazioni, l'adattamento ai canali e la gestione multicanale. Per l'entità Utente come chiave primaria si è scelta Utente(nome, cognome, email).

Robot:

L'entità Robot rappresenta le istanze fisiche o virtuali dei robot con cui gli utenti interagiscono. Il progetto è una piattaforma per la Human-Robot-Interaction ciò implica che il robot non è solo un software astratto ma anche un agente fisico o virtuale distinguibile che interagisce con gli umani, l'entità robot fornisce una rappresentazione concreta di questo fornendo coerenza con il dominio applicativo. Inoltre il testo menziona la personalizzazione e configurazione applicabile al sistema intelligente, l'entità robot è il punto di ancoraggio per tali configurazioni. Un robot avrà attributi specifici di un'istanza robotica, vitali per la manutenzione, aggiornamenti e allocazione delle risorse. Possiamo sapere anche le metriche di

performance di ogni robot attraverso il collegamento delle entità metrica→interazione→robot. Un sistema HRI reale difficilmente si limiterebbe a un singolo robot, quindi abbiamo deciso di renderlo il più realistico possibile, permettendo al database di supportare e gestire più istanze robotiche simultaneamente, ognuna con le proprie peculiarità e configurazioni, fondamentale per la scalabilità del progetto.

In sintesi l'entità robot è il pilastro per la gestione delle istanze fisiche o logiche degli agenti artificiali, essa abilita personalizzazione tracciabilità/gestione di errori e performance, l'analisi del dialogo, rendendo il sistema robusto, scalabile e allineato ai requisiti di un ambiente Human-Computer-Interaction. Senza l'entità robot, molte delle funzionalità avanzate descritte dal testo (profilazione dinamica, analisi performance di un modello specifico o diagnosi di errori hardware/software) sarebbero difficili da tracciare efficientemente nel database.

Interazione:

Questa è una delle entità centrali del database, che registra ogni singolo scambio comunicativo tra l'utente e il sistema intelligente. Ogni interazione ha diversi attributi che definiscono la data e ora esatta dell'interazione (cruciale per l'ordine cronologico e analisi temporale), il contenuto dell'input, ovvero la trascrizione dell'input fornito dall'utente, il tipo dell'input (la modalità con cui l'utente ha interagito, ad esempio visivo, vocale, tattile, testuale, ambientale...), e il contesto della conversazione, che fornisce informazioni sul contesto attuale del dialogo, essenziale per la memoria conversazionale estesa e la gestione delle conversazioni complesse. Interazione ha relazioni con sessione e robot, in quanto molte interazioni avvengono in una singola sessione e ogni interazione è parte di una sola sessione, questo è fondamentale per raggruppare gli scambi che compongono un dialogo continuo e mantenere il contesto. Ogni interazione è con un singolo robot, ciò permette di attribuire le interazioni a specifiche istanze robotiche.

Emozione:

L'entità emozione registra le emozioni rilevate (o generate) durante l'interazione tra utente e robot, basandosi sul modello circonflesso di Plutchik, fornendo un dettaglio analitico sulle emozioni. Ogni emozione rilevata ha un nome compreso tra le otto emozioni primarie identificate da Plutchik (paura, sorpresa, tristezza, disgusto, rabbia, aspettativa, gioia e fiducia) e le emozioni secondarie, per garantire che solo le emozioni valide possano essere inserite, mantenendo la coerenza e complessità con il

modello, un attributo 'intensità' definisce l'intensità dell'emozione rilevata, permettendo la modulazione del tono emotivo da parte del sistema intelligente. L'attributo 'categoria plutchik' permette di classificare le emozioni in base al loro livello di complessità nel modello (emozioni secondarie/primarie..), fondamentale per algoritmi di riconoscimento più avanzati. 'Emozione opposta' supporta un aspetto distintivo del modello di Plutchik che è l'organizzazione delle emozioni primarie in coppie di opposti polari (es. Gioia vs Tristezza ecc...), conoscere l'emozione opposta offre un'informazione preziosa per l'algoritmo di risposta, ad esempio se il robot rileva tristezza e deve guidare l'utente verso uno stato d'animo più positivo, può mirare a indurre risposte associate alla gioia; per la gestione di conversazioni complesse, se un utente manifesta un'emozione e poi rapidamente la sua opposta, potrebbe indicare un cambiamento rapido di stato che il sistema dovrebbe essere in grado rilevare e gestire; è utile anche per l'addestramento dei modelli, se un utente esprime un'emozione, il sistema può valutare se la sua risposta ha mosso l'utente verso l'emozione desiderata o se ha peggiorato la situazione.

Le relazioni tra emozione, risposta, suggerimento e interazione sono cruciali per supportare il modello di Plutchik nel contesto HRI:

- Le emozioni non esistono nel vuoto ma sono espresse e rilevate all'interno di un'interazione e spesso in relazione a specifici messaggi.
- Collegare emozione all'interazione e alla risposta/suggerimento del robot permette di capire quando e dove e in risposta a cosa l'emozione si è manifestata. Questo è fondamentale per: il riconoscimento multimodale avanzato (permette di associare l'emozione rilevata, tramite testo, voce, espressioni facciali, a una specifica interazione o risposta/suggerimento). Per l'apprendimento continuo, in quanto il sistema può apprendere i pattern emotivi in base al contesto, affinando la sua capacità di riconoscimento e generazione di risposte emotive.

L'entità emozione è stata modellata per riflettere e sfruttare appieno la ricchezza del modello circonflesso di Plutchik.

Sessione:

Rappresenta una sessione continua di interazioni tra utente e robot. Essa è fondamentale per la gestione del contesto conversazionale e per sessioni sicure. Token supporta l'accesso sicuro, essa è la credenziale che il client userà per comunicare con il server, garantendo che solo gli utenti autenticati e le sessioni valide possano

interagire con il robot, il suo ruolo è sicurezza e validazione e di identificazione interna del record.

Ogni sessione ha una data e ora di inizio e di fine, un token.

Un utente può avviare molte sessioni, ma ogni sessione è avviata da un singolo utente.

Ogni sessione può contenere più interazioni, ma ogni interazione è contenuta in una singola sessione.

Risposta e Suggerimento:

Le tabelle risposta e suggerimento contengono le risposte/suggerimenti che il sistema intelligente fornisce. Ogni risposta/suggerimento è associata/o ad una o più emozioni e contesti in modo tale che il sistema intelligente possa rendere il più emotivamente coinvolgente la conversazione. Queste entità sono fondamentali per la generazione di risposte adeguate e per la modulazione del tono. Un robot fornisce sempre almeno una risposta all'input dell'utente e sempre almeno un suggerimento, questa scelta di fornire sempre almeno un suggerimento, invece che renderlo facoltativo, è stata fatta appositamente per rendere la comunicazione, tra essere umano e robot, più coinvolgente e mirata alle specifiche del sistema, che richiedono, che il sistema intelligente possa persuadere l'utente a scelte etiche e per il benessere della propria salute e/o di quella altrui.

Ogni risposta e/o suggerimento sono caratterizzati da:

- un contesto conversazionale in cui il messaggio è stato generato,
- l'intensità dell'emozione dell'utente, in modo da regolare il tono e il linguaggio,
- il timestamp che definisce il momento esatto in cui la risposta è stata generata,
- la condizione per la cui fornire il suggerimento,
- il contenuto della risposta,
- il tipo di tale output,

Ogni robot può generare una o più risposte e suggerimenti, ma ogni risposta e suggerimento appartiene ad un unico robot.

Competenza Robot e Personalizzazione Robot:

Il dominio del database è la Human-Robot-Interaction, essa pone forte attenzione alla personalizzazione e alla capacità dei robot di adattarsi a diversi utenti e contesti; per

tali motivi abbiamo scelto di introdurre le entità: Competenza robot e Personalizzazione robot.

L'entità competenza robot rappresenta le caratteristiche e le capacità intrinseche del robot,

mentre personalizzazione robot rappresenta le configurazioni e parametri personalizzabili dall'utente che vengono applicati a un robot in un certo contesto.

Nello schema E-R non ristrutturato si è deciso di mantenere i due concetti separati perchè uno descrive le caratteristiche del robot (es. capacità multimodale, livello autonomia, supporto linguistico ecc...) mentre la personalizzazione descrive adattamenti e configurazioni dinamiche, come parametri di personalizzazione scelti da un utente, preferenze attivate/disattivate, comportamenti personalizzati. Inoltre mantenerle separate migliora la chiarezza e leggibilità del modello.

Queste entità rispondono al requisito di personalizzazione avanzata e alla necessità di adattamento del sistema, permettendo agli utenti di influenzare il comportamento e le modalità operative del robot.

La tabella competenza robot è caratterizzata da diversi attributi:

- nome competenza, che definisce il nome della competenza del robot,
- ultima calibrazione, timestamp che definisce la data di ultima calibrazione della competenza, per sapere quando è stata ottimizzata l'ultima volta,
- descrizione, una descrizione testuale della competenza, fornisce dettagli tecnici,
- numero utilizzi, tiene traccia di quante volte è utilizzata tale competenza, permette di monitorare la frequenza di utilizzo e la popolarità della competenza,
- categoria competenza, categoria funzionale utile per raggruppare e classificare le competenze per scopi di analisi e report,
- livello competenza, grado o livello di abilità, serve per differenziare robot più sofisticati o per determinare la qualità delle risposte,
- algoritmi utilizzati, lista degli algoritmi utilizzati (es. deep learning, NLP ecc...), aiuta gli sviluppatori a capire la tecnologia alla base della competenza,
- stato competenza, stato attuale (es. attiva, non attiva, in manutenzione...), permette di sapere se la competenza è attiva o va riattivata,
- data acquisizione, data in cui la competenza è stata acquisita per il robot, serve per sapere quando la competenza è diventata disponibile.

Alcuni attributi sono descrittivi mentre altri di monitoraggio.

Per quanto riguarda l'entità Personalizzazione robot abbiamo gli attributi:

- attiva, un flag che indica se la personalizzazione è attualmente attiva o meno,
- priorità, valore per indicare la priorità della personalizzazione, se ci sono conflitti tra parametri la priorità aiuta a decidere quali parametri hanno la precedenza,
- tipo personalizzazione, descriva la tipologia di personalizzazione (es. emotiva, comportamentale, visiva ecc...), permette di classificare e filtrare le personalizzazioni,
- nome parametro, il nome del parametro specifico che viene personalizzato (es. tono voce, velocità risposta ecc...), consente flessibilità e modularità della risposta,
- valore parametro, valore effettivo assegnato al parametro, questo è ciò che rende la personalizzazione operativa,
- livello applicazione, indica dove o a quale livello è stata applicata la personalizzazione (es. utente, canale, globale ecc...)
- data applicazione, data in cui la personalizzazione è stata applicata o modificata, permette di tenere traccia della cronologia delle personalizzazioni e la loro evoluzione nel tempo.

Accessibilità:

Questa entità è dedicata alla memorizzazione delle impostazioni di accessibilità personalizzate per ciascun utente. Garantisce che il sistema sia inclusivo e utilizzabile da tutti gli utenti, compresi quelli con esigenze speciali. Permette al robot di adattare l'output (es. dimensioni testo, contrasto ecc...) e l'input (es. navigazione tastiera...) in base alle necessità individuali.

Come attributi troviamo:

- contrasto alto, abilita/disabilita un tema a contrasto elevato, cruciale per utenti con problemi di vista,
- dimensioni testo, controlla la grandezza del testo (es. piccolo, normale, grande...), supportando leggibilità per diverse esigenze visive,
- data configurazione, traccia l'ultima data in cui le impostazioni di accessibilità sono state modificate, utile per capire l'evoluzione delle preferenze,
- riduzione animazioni, per utenti sensibili a movimenti rapidi o animazioni complesse,

- navigazione tastiera, permette la navigazione completa dell'interfaccia tramite tastiera, essenziale per utenti con disabilità motorie o che preferiscono non usare il mouse,
- sottotitoli audio, abilita i sottotitoli per contenuti audio e video,
- lingua segni, indica la preferenza per il supporto alla lingua dei segni, suggerendo al sistema di fornire interpretazioni visive ove possibile,
- supporto screen reader, abilita funzionalità specifiche per lettori di schermo,
- note speciali, campo libero per annotazioni aggiuntive su esigenze specifiche non coperte da altri campi.

Ogni utente ha un'unica configurazione accessibilità. Separare le impostazioni di accessibilità in una tabella dedicata, piuttosto che come colonne nella tabella utente, promuove la normalizzazione del database. Mantiene la tabella utente più snella, raggruppando informazioni correlate ad un'unica entità. Inoltre permette una gestione più flessibile di queste impostazioni e un'eventuale evoluzione delle opzioni di accessibilità.

Configurazione Privacy:

Questa entità gestisce le preferenze di privacy e i consensi dell'utente riguardo al trattamento dei dati personali e alle comunicazioni marketing. Essa è cruciale per la conformità alla normativa (es. GDPR).

Attributi:

- consenso dati personali: indica se l'utente ha dato il consenso al trattamento dei dati personali,
- consenso marketing, consenso per ricevere comunicazioni marketing,
- consenso profilazione, consenso per l'utilizzo dei dati per la profilazione (es. suggerimenti personalizzati...),
- data consenso, la data in cui i consensi sono stati forniti,
- data scadenza, la data di scadenza del consenso, importante per la conformità legale.

Ogni utente deve configurare le impostazioni di privacy al primo accesso, quindi tale operazione è obbligatoria. Come per accessibilità e profilo utente, separare le impostazioni di privacy in una tabella dedicata serve a normalizzare il database e mantenere la tabella utente pulita. Questo permette di gestire i complessi requisiti legali e di consenso in modo indipendente e specifico.

Profilo Utente e Preferenze Utente:

Queste entità memorizzano i dettagli dinamici e psicografici sul profilo dell'utente, come tratti di personalità, stato emotivo prevalente e stile comunicativo, oltre a preferenze personalizzabili.

Attributi di profilo utente:

- tratti personalità, descrizione dei tratti della personalità dell'utente,
- stato emotivo prevalente, l'emozione o lo stato emotivo più frequentemente rilevato per l'utente,
- stile comunicativo, il modo in cui l'utente preferisce comunicare (es. formale, informale, uso di emoji...),
- ultima modifica, data dell'ultima modifica del profilo,

Attributi di preferenze utente:

- nome preferenza, nome di una preferenza specifica, deve essere univoco,
- attiva, stato di attivazione della preferenza,
- descrizione, descrizione dettagliata della preferenza,
- data modifica, la data dell'ultima modifica della preferenza,
- tipo, la tipologia della preferenza,
- valore preferenza, dà alla preferenza un valore.

Il motivo per cui abbiamo scelto di mantenere una relazione 1:N tra utente e profilo utente, nonostante sia controintuitivo, è perché essa è una decisione sensata in un modello per un sistema HRI avanzato. Una persona può avere diversi profili psicologici nel tempo, magari per diversi lassi di tempo, si può sentire in modi diversi. In questo modo si ha un profilo dinamico e multidimensionale, che non riflette i dati statici ma anche aspetti dinamici, come stati d'animo ricorrenti... Se un utente cambia stato d'animo nel tempo (cosa realistica) avere più profili psicologici permette di storicizzare e tracciare queste evoluzioni.

Inoltre il sistema potrebbe generare e salvare nuovi profili basati sull'interazione per affinare la sua comprensione dell'utente. Avere un profilo per ogni fase di apprendimento è possibile con una relazione 1:N. Ulteriormente questo approccio aiuta l'adattamento contestuale, ad esempio un utente potrebbe avere un profilo 'da lavoro' (più formale e orientato al compito) e uno 'personale' (più empatico e orientato al benessere). Questa struttura permette di analizzare come il profilo psicologico cambia nel tempo.

Per quanto riguarda le preferenze utente, un utente può avere 0 o più preferenze, ed una preferenza appartiene ad un solo utente.

Contenuto Multimediale:

L'entità contenuto multimediale memorizza i riferimenti a contenuti multimediali (video, immagini, audio, testo, animazioni, vr...) che possono essere parte di un'interazione, permettendo la gestione dei contenuti. Contenuto multimediale non è stata integrata direttamente come attributo di interazione per diverse ragioni, che portano a numerosi vantaggi per il sistema HRI.

Un'interazione realistica, di un sistema HRI che è intrinsecamente multimodale, non si limita a un singolo tipo di contenuto multimediale. Una risposta del robot o un input dell'utente potrebbe includere contemporaneamente testo, audio, immagini, video,

contenuti interattivi ecc... Ogni tipo di contenuto multimediale ha i suoi attributi specifici. Trovare tutti i video o tutte le immagini usate nelle interazioni diventerebbe più difficile e meno efficiente, richiedendo query complesse su molte colonne diverse, inoltre includere contenuto multimediale in interazione violerebbe la prima forma normale (se un'interazione potesse avere più valori per un singolo attributo si violerebbe la 1NF, che richiede che ogni colonna contenga valori atomici).

Attributi:

- tipo contenuto, specifica la natura del contenuto (es. video, immagine, audio, testo ecc...),
- data caricamento, la data e l'ora del caricamento,
- percorso file, il path o URL dove è memorizzato il file multimediale,
- descrizione, una descrizione testuale del contenuto.

Molti contenuti multimediali possono essere associati a una singola interazione, ma ogni contenuto appartiene a una sola interazione. In quanto un'interazione può coinvolgere più elementi multimediali.

Feedback:

L'entità feedback cattura le valutazioni e i commenti degli utenti riguardo alle interazioni con il sistema. Questa entità è essenziale per l'apprendimento automatico e il miglioramento delle capacità del robot.

Attributi:

- valutazione, punteggio numerico del feedback, cruciale per l'analisi quantitativa,
- commento, testo libero del commento dell'utente, fornendo dettagli qualitativi,
- timestamp, la data e l'ora in cui il feedback è stato lasciato.

Un utente non è obbligato a lasciare un feedback, ma può lasciarne solo uno per ogni interazione. Una singola interazione può ricevere 0 o 1 feedback, ogni feedback riferisce a una singola interazione.

Questa struttura permette di tracciare chi ha lasciato il feedback e a quale specifica interazione si riferisce.

Interfaccia canale:

Definisce le caratteristiche e le configurazioni di ciascun canale o interfaccia di comunicazione supportato dal sistema (es. chatbot, app mobile, interfaccia vocale ecc...), Fondamentale per il supporto multimodale.

Attributi:

- tipo canale, la natura fisica o logica del canale (es. visivo, uditivo, tattile, olfativo, propriocettivo...),
- ultima connessione, data e ora dell'ultima attività su quel canale, utile per monitoraggio e debugging,
- modalità output supportate, tipi di output che il canale può generare (es. visivo, vocale...),

- modalità input supportate, tipi di input che il canale può ricevere (es. visivo, vocale, testuale...),
- parametri autenticazione, tipo di meccanismo di autenticazione supportato dal canale (es. password, token, mfa...),
- configurazione json, un campo JSON che memorizza configurazioni flessibili per il canale (es. lingua, volume voce, sensibilità touch...), questa è una scelta di design moderna che consente grande flessibilità senza alterare lo schema del database per ogni nuova opzione di configurazione,
- nome canale, un nome descrittivo per il canale (es. app mobile, interfaccia vocale...),
- url endpoint, l'url o l'endpoint di rete per la comunicazione con il canale,
- attivo, flag che indica se il canale è attualmente operativo,
- data creazione, data di creazione del record del canale.

La tabella Interfaccia canale è stata progettata per rappresentare i vari canali di comunicazione disponibili nel sistema. La relazione molti a molti con utente consente di personalizzare e abilitare canali diversi per ciascun utente, mentre la relazione molti a molti con interazione permette di tracciare i canali effettivamente utilizzati durante ogni interazione, in quanto ogni interazione avviene su uno o più canali (ad esempio: la stessa conversazione può avvenire contemporaneamente via voce e testo, oppure l'utente può cambiare canale a metà conversazione...).

Quindi ogni utente utilizza una o più interfacce, ma una interfaccia non è necessariamente usata da un utente o un'interazione. Un'interfaccia però può essere usata da massimo N utenti e N interazioni.

Log Errori:

Questa entità funge da registro centrale per tutti gli errori e le eccezioni che si verificano nel sistema robotico e nelle interazioni con l'utente. Supporta il debugging, la manutenzione e il monitoraggio della stabilità del sistema.

Attributi:

- codice errore, un codice standardizzato per l'errore,
- livello gravità, l'importanza o l'impatto dell'errore,
- azione correttiva, un suggerimento su come risolvere l'errore,
- stack trace, il dettaglio tecnico dell'errore, essenziale per il debugging,
- messaggio errore, una descrizione più leggibile dell'errore (es. connessione database fallita...),
- timestamp, la data e l'ora in cui si è verificato l'errore,
- componente sistema, la parte del sistema che ha generato l'errore (es. database, API gateway, NLP engine...),
- tecnico assegnato, il nome del tecnico responsabile della risoluzione,
- stato risoluzione, fornisce informazioni sullo stato di risoluzione del problema (es. risolto, in manutenzione ecc...).

Una sessione può generare 0 o più errori ma un errore è di al massimo una sessione.

Un robot può generare anch'esso 0 o più errori ma un errore è legato al massimo a un robot. Tuttavia un singolo log errore potrebbe non essere associato a nessun robot o

nessuna sessione specifica, ad esempio nel caso di un errore globale (un errore di configurazione nel database o nel servizio di intelligenza artificiale potrebbe impedire l'avvio di qualsiasi sessione o robot)

Metrica:

L'entità metrica raccoglie i dati statistici e di performance sulle interazioni e sul sistema. Essa è cruciale per l'apprendimento continuo, fornendo i dati quantitativi necessari per valutare l'efficacia del sistema.

Attributi:

- nome, il nome della metrica,
- valore metrica, il valore numerico della metrica,
- descrizione, una descrizione della metrica,
- timestamp, la data e l'ora della misurazione,
- tipo metrica, la categoria della metrica (es. performance, accuratezza, affidabilità, precisione...),
- errore, specifica se è stato rilevato un errore in concomitanza con la metrica,
- numero errori, quanti errori specifici sono associati a questa misurazione.

Molte metriche possono essere calcolate per una interazione (es. tempo di risposta, accuratezza di riconoscimento vocale, accuratezza NLP per la stessa interazione ecc..), ma ogni metrica si riferisce a una sola interazione.

Policy Moderazione:

Definisce le regole e le azioni per la moderazione di contenuti inappropriati o non conformi. Questa entità è un componente cruciale per la gestione della sicurezza e della privacy, e per garantire che le interazioni siano etiche e sicure.

Attributi:

- descrizione, una spiegazione dettagliata dello scopo e delle condizioni della policy,
- azione, l'azione che il sistema deve intraprendere quando la policy è violata,
- tipo contenuto, a quale categoria di media la policy si applica (es. testo, audio, comportamento...),
- attiva, indica se la policy è attualmente in vigore.

Le policy di moderazione vengono applicate ad ogni singolo messaggio. Quindi ogni messaggio viene filtrato in modo tale che si accerti che esso sia conforme con le normative vigenti.

Storico Emotivo:

Questa entità è fondamentale e ha una funzione centrale nel modello, in quanto grazie al suo compito, cioè la registrazione e storicizzazione delle emozioni, il robot può adattarsi meglio alle esigenze dell'utente.

Attributi:

- Timestamp, che identifica il momento in cui l'emozione viene storicizzata,
- Emozione prevalente, rappresenta l'emozione che viene registrata più volte,
- intensità, l'intensità dell'emozione registrata.
- emozioni, le emozioni precedentemente registrate

Uno storico emotivo è generato da un unico utente ed un utente genera un unico storico emotivo. Mentre un'emozione è registrata una o più volte nello storico emotivo, in quanto emozione non è una tabella statica di emozioni ma una tabella che registra le emozioni rilevate in una interazione con un utente, e uno storico emotivo possiede una o più emozioni.

Traduzione:

La funzionalità di traduzione è essenziale per un sistema multilingua. Questa funzione è gestita dall'entità traduzione.

Attributi:

- lingua, lingua in cui deve essere tradotta la risposta,
- testo tradotto, la risposta tradotta nella lingua di riferimento,

Ogni risposta può avere più traduzioni, una traduzione è relativa ad una risposta.

Pattern di progettazione:

Un pattern di progettazione fondamentale applicato in questo modello è la

Reificazione degli Attributi. Questo pattern consiste nel trasformare un attributo o un gruppo di attributi da una semplice colonna in un'entità a sé stante all'interno del modello. La reificazione è stata applicata quando un attributo aveva propri attributi descrittivi, poteva essere riusato o condiviso, la sua registrazione è storicizzata o contestualizzata e può avere molteplici occorrenze.

Esempio: il concetto di *emozione* non è stato trattato come semplice attributo di “interazione”, ma è stato modellato come **entità** con attributi propri (nome, descrizione, intensità) e connessa a più entità (interazione, risposta, suggerimento, storico emotivo).

Lo stesso per il concetto di metrica e contenuto multimediale che non sono stati integrati come attributi in interazione, grazie alla loro complessità descrittiva.

Stessa cosa vale per profilo utente, accessibilità e configurazione privacy, che non sono stati trattati come semplici attributi di utente.

Strategia di progetto utilizzata:

La strategia di progettazione adottata per questo modello è un approccio misto, che combina elementi **Top-Down** e **Bottom-Up**. Questo metodo ibrido ha permesso di affrontare le complessità del dominio HRI in modo efficace.

Fase Top-Down:

L'approccio top-down ha dominato la fase iniziale di definizione del modello.

Partendo da una visione complessiva del sistema HRI, gli obiettivi generali sono stati scomposti progressivamente in requisiti funzionali e non funzionali dettagliati.

Si è partiti dalla comprensione delle macro funzionalità necessarie. Successivamente, sono state identificate le entità centrali del dominio (es. robot, utente, interazione, emozione ecc...), queste entità rappresentano i concetti chiave che il database deve gestire. Le interconnessioni tra queste entità sono state stabilite, definendone le cardinalità logiche (es. un utente avvia molte sessioni, una sessione ha molte interazioni ecc...).

L'adozione del modello Top-Down ha garantito che il modello fosse allineato con gli obiettivi del sistema HRI, focalizzandosi sulle entità più importanti per il funzionamento complessivo della piattaforma.

Fase Bottom-Up:

L'approccio Bottom-Up è emerso con forza nella fase di raffinamento. Una volta stabilito lo scheletro principale dello schema, l'attenzione si è spostata sulle specifiche esigenze di ogni componente. Ogni entità è stata arricchita con attributi specifici, considerando i dati minimi per supportare i requisiti. La reificazione di alcuni attributi, dettata dalla necessità di maggiore dettaglio e flessibilità per i dati che all'inizio potevano sembrare semplici attributi (es. emozione, metrica...). La definizione di tabelle come log errori e policy moderazione, con i loro attributi dettagliati e relazioni, è un esempio di come le esigenze di monitoraggio e sicurezza abbiano influenzato la creazione di entità specifiche. L'integrazione di tipi di dato complessi come JSON o TEXT per campi descrittivi estesi è una decisione Bottom-Up per raccogliere dati non strutturati.

L'approccio Bottom-Up ha permesso di affinare il modello, aggiungendo il livello di dettaglio necessario per implementare pienamente le funzionalità richieste, garantendo al contempo flessibilità e adattabilità.

Regole aziendali

Vincoli di Integrità:

RV1	Ogni profilo utente deve essere associato a un esatto utente.
RV2	Ogni interazione deve essere associata a una sessione esistente
RV3	Ogni interazione può essere associata a una o più emozioni.
RV4	Ogni risposta può essere collegata a una o più emozioni.
RV5	Ogni risposta deve essere generata da un solo robot.
RV6	Ogni feedback deve essere fornito da un solo utente e riferito a una sola interazione.
RV7	Ogni robot deve possedere almeno una competenza.
RV8	Ogni contenuto multimediale deve essere associato a una sola interazione.
RV9	Ogni utente può fornire al massimo un feedback per ciascuna interazione.
RV10	Ogni traduzione deve riferirsi a una sola risposta.
RV11	Ogni suggerimento può essere associato a una o più emozioni.
RV12	Ogni personalizzazione del robot deve essere eseguita da un utente e deve essere riferita a un robot specifico.
RV13	Ogni metrica deve essere collegata a un'interazione e avere dati misurabili (ad esempio tempo > 0).
RV14	Ogni configurazione di accessibilità deve essere associata a un solo utente.
RV15	Ogni politica di moderazione è applicata a ogni risposta e/o suggerimento.
RV16	Un utente può avviare una sessione solo se autenticato correttamente.
RV17	Ogni emozione rilevata deve essere associata ad un'interazione in cui è stata rilevata.
RV18	Ogni interazione deve essere associata ad almeno un canale di comunicazione valido.
RV19	Una risposta può essere tradotta solo se è stata generata.
RV20	Ogni suggerimento deve essere generato da un robot e ogni robot genera almeno un suggerimento, affinché possa supportare l'interazione in modo attivo e naturale.
RV21	Ogni feedback deve essere riferito a una interazione già conclusa.
RV22	L'intensità delle emozioni rilevate deve essere un valore numerico compreso tra 0.00 e 1.00, coerente con la scala di Plutchik.

RV23	Solo le emozioni primarie e secondarie di Plutchik possono essere registrate come nome nell'entità Emozione.
RV24	Emozioni opposte non devono coesistere nella stessa interazione.
RV25	Un emozione è associata ad un'interazione e/o suggerimento.

Derivazioni:

RD1	Il tempo medio di risposta per utente si ottiene calcolando la media dei tempi nelle metriche associate alle sue interazioni.
RD2	Lo stato emotivo prevalente dell'utente si ottiene analizzando le emozioni più frequenti nello storico emotivo.
RD3	Il numero di interazioni per sessione si ottiene contando le interazioni collegate a ciascuna sessione.
RD4	La lingua preferita dell'utente può essere derivata dalla lingua più frequente tra le traduzioni ricevute.
RD5	Il tasso di errore del robot si ottiene dividendo il numero di log errori per il numero totale di interazioni in cui è coinvolto.
RD6	Il tasso di feedback positivi si ottiene calcolando la proporzione di valutazioni superiori a una soglia (ad esempio > 7) tra tutti i feedback ricevuti.
RD7	Il canale preferito dall'utente si ottiene identificando il canale più utilizzato nelle sue interazioni.
RD8	L'efficacia delle personalizzazione del robot può essere stimata confrontando le metriche (tempo di risposta, feedback..ecc.) prima e dopo la personalizzazione.
RD9	La frequenza delle emozioni per utente si ottiene conteggiando il numero di volte che ogni emozione compare nel suo storico emotivo.
RD10	Il tipo di risposta da fornire può essere derivato dalla combinazione di due emozioni primarie rilevate nella stessa sessione secondo il modello Plutchik.
RD11	Il numero medio di feedback per utente si ottiene calcolando il rapporto tra numero di feedback forniti e numero totale di interazioni.
RD12	Il livello di coinvolgimento emotivo di un'interazione si può calcolare come media pesata dell'intensità delle emozioni rilevate.
RD13	Le performance del robot personalizzato si ottengono confrontando le metriche con la configurazione di personalizzazione attiva con quelle con la personalizzazione disattivata.

RD14	Il tasso di moderazione attiva si calcola dividendo il numero di contenuti soggetti a policy per il totale di risposte e suggerimenti generati.
RD15	Il livello di coinvolgimento emotivo di un utente si ottiene calcolando la media delle intensità delle emozioni nel suo storico emotivo.

Tabella Operazioni - Requisiti:

Operazione	Requisito collegato	Entita' coinvolte
Op1: registrare un nuovo utente con profilo, preferenze, lingua, privacy e accessibilità.	Il sistema deve registrare un nuovo utente, memorizzare il profilo utente, associare a ciascun utente una configurazione di accessibilità, una configurazione privacy, una lingua preferita e preferenze personali.	Utente, Profilo_Utente, Preferenze_Utente, Config_Privacy, Accessibilità.
Op2: Interazione utente-robot tramite interfaccia canale.	Il sistema deve supportare l'interazione tra utente e robot, deve gestire le interazioni attraverso diverse interfacce canale, deve consentire l'avvio, la gestione delle sessioni utente, e deve registrare i contenuti multimediali associati alle interazioni.	Utente, Robot, Interazione, Interfaccia_Canale.
Op3: Rilevazione emozioni, intensità e feedback in interazioni.	Il sistema deve rilevare le emozioni espresse dall'utente durante l'interazione, associare ad ogni emozione rilevata un livello di intensità, consentire all'utente di fornire feedback esplicito, memorizzare il feedback associandolo alla risposta ricevuta.	Interazione, Emozione, Feedback, Utente.
Op4: Genarazione di risposte e suggerimenti basati su emozione e contesto.	Il sistema deve generare una risposta coerente con l'emozione rilevata, deve fornire suggerimenti all'utente in base al contesto emotivo, deve memorizzare le competenze possedute dal robot, associare ad ogni robot una configurazione personalizzata.	Robot, Emozione, Risposta, Suggerimento, Interazione, personalizzazione_robot, competenze_robot.
Op5: Traduzione e adattamento linguistico delle risposte.	Il sistema deve memorizzare le traduzioni delle risposte in più lingue, associare a ciascun utente una lingua preferita. Le risposte devono essere emotivamente appropriate, coerenti con l'emozione	Risposta, Traduzione, Utente, Emozione.

	rilevata e tradotte nella lingua dell'utente.	
Op6: Competenze del robot, comprensione input, adattamento allo stile dell'utente.	Il sistema deve dare risposte emotivamente appropriate, coerenti con l'emozione rilevata e tradotte nella lingua dell'utente, deve generare una risposta coerente con l'emozione rilevata, fornire suggerimenti all'utente in base al contesto emotivo e culturale, e gestire interazioni attraverso diverse interfacce canale.	Robot, Emozione, Storico_Emotivo, Competenza_Robot, Profilo_Utente, Traduzione, Suggerimento, Risposta, Interfaccia_Canale
Op7: Personalizzazione del robot da parte dell'utente.	Il sistema deve memorizzare la configurazione di personalizzazione del robot e memorizzare le competenze possedute dal robot.	Robot, Personalizzazione_Robot, Competenze_Robot, Utente.
Op8: Archiviazione e ricerca delle interazioni.	Il sistema deve memorizzare le interazioni tra utente e robot, monitorare metriche di utilizzo del sistema.	Interazione, Utente, Storico_Emotivo, Robot, Log_Errori, Metrica, Sessione.
Op9: Apprendimento da feedback esplicito e implicito.	Il sistema deve apprendere dai feedback espliciti forniti dall'utente, monitorare metriche di utilizzo del sistema, registrare eventuali errori generati durante l'interazione.	Feedback, Metrica, Storico_Emotivo, Robot, Log_Errori.
Op10: Moderazione dei contenuti generati e ricevuti.	Il sistema deve applicare politiche di moderazione per evitare contenuti inappropriati e operare secondo principi etici e trasparenti.	Policy_Moderazione, Risposta, Suggerimento.
Op11: Gestione delle sessioni, autenticazione e sicurezza.	Il sistema deve associare a ciascun utente una configurazione di privacy, garantire la sicurezza e la protezione dei dati dell'utente. Il sistema deve essere progettato con un approccio centrato sull'utente, rispettando i principi etici.	Sessione, Utente, Config_Privacy, Accessibilità, Log_Errori, Robot, Interfaccia_Canale.
Op12: Traduzione input/output e supporto multilingua.	Il sistema deve memorizzare le traduzioni delle risposte in più lingue e associare a ciascun utente una lingua preferita.	Traduzione, Risposta, Interazione, Utente.

PROGETTAZIONE LOGICA

2.1. Eliminazione delle generalizzazioni

Nel nostro **schema E-R iniziale** non sono state incluse generalizzazioni esplicite. Questa decisione è stata presa fin dalle prime fasi di modellazione con un approccio mirato alla **chiarezza e alla fedeltà immediata al dominio applicativo**.

1. **Priorità alla Mappatura Fedele del Dominio:** L'intento primario in fase concettuale era quello di rappresentare esplicitamente e senza ambiguità tutti i concetti e le funzionalità richieste dai requisiti del progetto. Evitare generalizzazioni ha permesso di mappare direttamente le entità così come erano percepite nel contesto HRI.
2. **Chiarezza per la Traduzione in SQL:** Un modello E-R più diretto e meno astratto semplifica notevolmente il passaggio alla fase di progettazione logica e alla successiva implementazione SQL. Questo ha ridotto la complessità di interpretazione e la probabilità di errori nella traduzione del modello.
3. **Differenziazione Funzionale Esplicita:** Concetti come "Risposta" (reattiva) e "Suggerimento" (proattivo/persuasivo) sono stati inizialmente considerati come entità distinte nello schema non ristrutturato. Questo ha garantito che le loro diverse funzioni e comportamenti, citati esplicitamente nel testo del progetto, fossero pienamente riconosciuti prima di qualsiasi ottimizzazione strutturale. La fase concettuale ha il compito di mappare i concetti espliciti così come sono, e abbiamo creduto fosse meglio mantenere entità separate per mappare fedelmente il dominio, per poi ottimizzare nella fase di ristrutturazione.

2.1.2. Partizionamento/accorpamento di entità e associazioni

La fase di ristrutturazione ha comportato l'applicazione di principi di normalizzazione e ottimizzazione volti a consolidare lo schema E-R iniziale, rendendolo più efficace e manutenibile senza perdere la ricchezza concettuale.

1. Il primo accorpamento che abbiamo deciso di fare è:

- **Risposta + Suggerimento → Messaggio**

Motivazioni:

Omogeneità Strutturale e Semantica: Sia le risposte che i suggerimenti sono essenzialmente stringhe di testo (o output multimodali) generate dal robot.

Condividono attributi comuni come il testo, la lingua, il tono e il contesto. Sebbene il loro *comportamento* (reattivo vs. proattivo/persuasivo) sia differente, la loro *rappresentazione dati* è quasi identica.

Prevenzione della Duplicazione di Schemi: Mantenere entità separate avrebbe comportato la creazione di due tabelle quasi identiche nel database relazionale, con un'inevitabile ridondanza nella gestione e nella manutenzione.

Semplicità del Modello Logico: L'accorpamento in un'unica entità messaggio con un attributo discriminante 'tipo_messaggio' (risposta o suggerimento) semplifica la progettazione logica e le query future. Invece di dover unire due tabelle diverse, tutte le informazioni sui messaggi del robot sono accessibili da un'unica fonte.

Flessibilità per Estensioni Future: Questa struttura è più flessibile per l'introduzione di nuovi tipi di messaggi (es. avviso ecc...) senza la necessità di creare nuove tabelle o alterare significativamente lo schema esistente.

2. Il secondo accorpamento:

- **Preferenze_Utente + Profilo_Utente → Profilo_Utente**

Motivazioni:

Coesione delle Informazioni Utente: Entrambe le entità preferenze_utente e profilo_utente erano strettamente legate all'utente e contenevano dati dinamici e personalizzabili (tratti di personalità, stato emotivo prevalente, stile comunicativo, attivazione di preferenze). L'accorpamento in una singola entità 'profilo_utente' ha consolidato tutte le informazioni relative al profilo dinamico e alle preferenze personalizzate dell'utente in un unico punto logico e fisico.

Relazione 1:N con Utente: Come discusso, la decisione di permettere a un utente di avere "uno o più profili psicologici" ha ulteriormente giustificato questo accorpamento. La fusione in profilo_utente con una relazione 1:N con utente riflette meglio la natura "dinamica" e "multidimensionale" del profilo, come richiesto, dove un utente può avere diverse "tendenze" del suo profilo nel tempo.

3. Il terzo accorpamento:

- **Competenza_Robot + Personalizzazione_Robot → Configurazione_Robot**

Motivazioni:

Natura Simile delle Proprietà Configurabili: Sia le "competenze" (cosa il robot sa fare, es. riconoscimento vocale, generazione di testo) che le "personalizzazioni" (come il robot si adatta all'utente, es. tono di voce, velocità di risposta) rappresentano forme di configurazione del comportamento o delle capacità del robot.

Attributi Condivisi: Molti attributi potenziali (es. nome del parametro, valore, data di applicazione) sarebbero stati simili o duplicati in due tabelle separate.

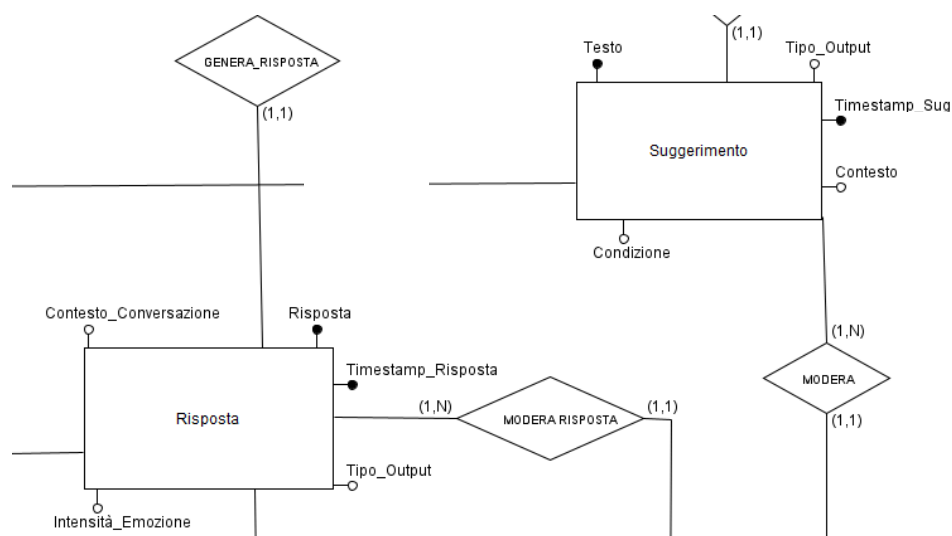
Flessibilità per Estensioni Future: L'entità configurazione_robot consolidata, con un attributo tipo_configurazione (es. 'competenza' o 'personalizzazione'), offre una maggiore flessibilità per aggiungere nuove categorie di configurazioni o parametri specifici del robot senza la necessità di alterare lo schema principale. Questo allineamento supporta il requisito di un robot "personalizzato dall'utente" e con "competenze" adattabili.

Semplicità di Gestione: Centralizzare tutte le configurazioni del robot semplifica la gestione, il monitoraggio e l'applicazione delle impostazioni, migliorando l'efficienza operativa.

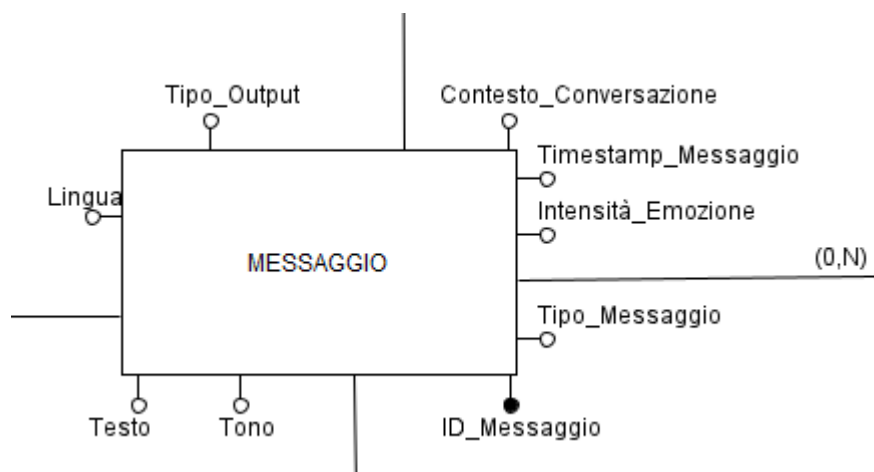
2.1.2.1. Accorpamenti:

Accorpamento 1 (Risposta + Suggerimento → Messaggio):

Porzione di schema prima dell'accorpamento:

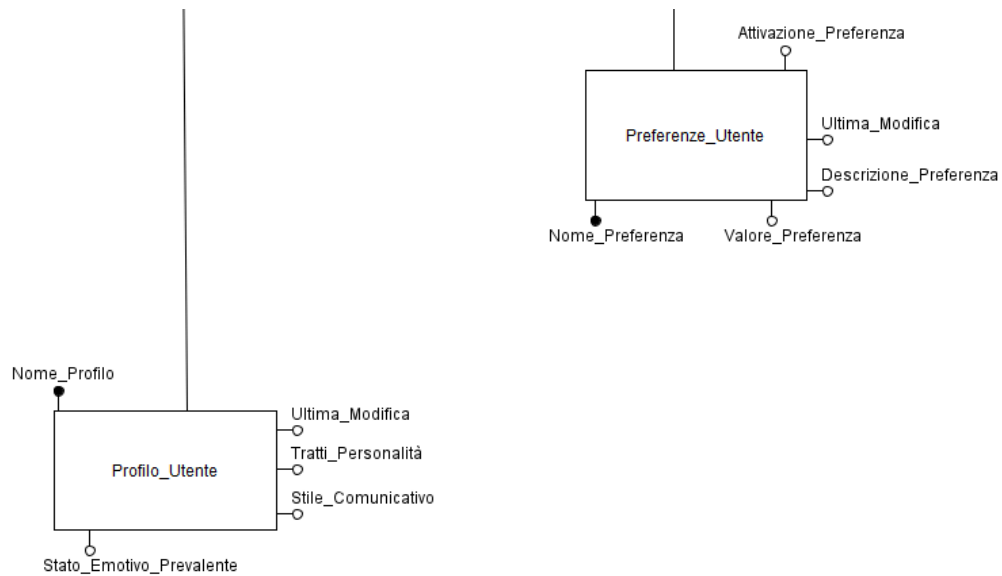


Porzione di schema dopo l'accorpamento:

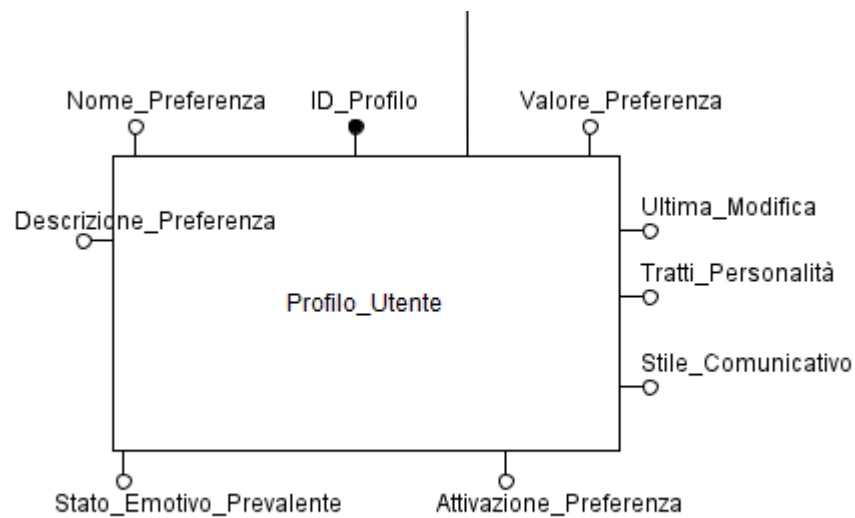


Accorpamento 2 ($\text{Preferenze_Utente} + \text{Profilo_Utente} \rightarrow \text{Profilo_Utente}$):

Porzione di schema prima dell'accorpamento:

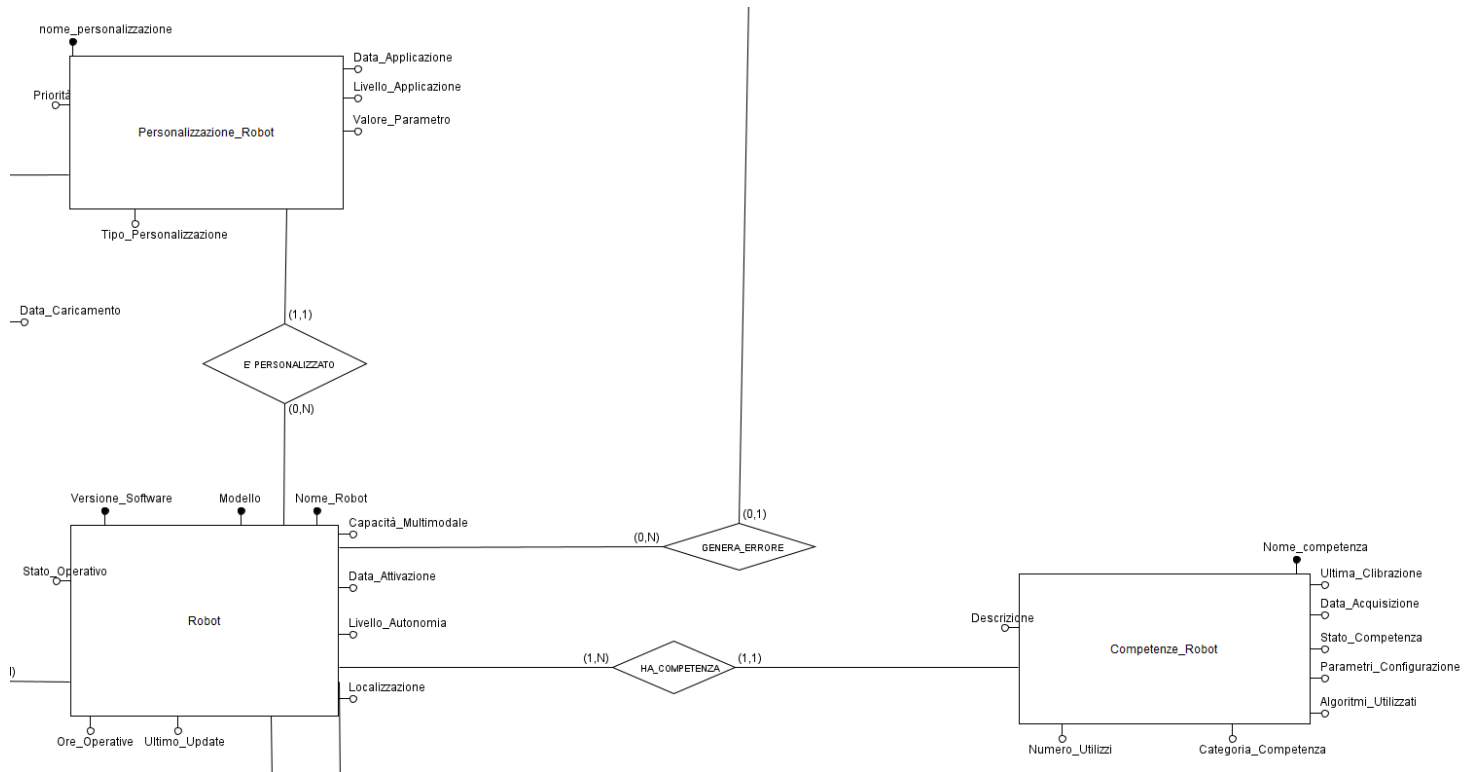


Porzione di schema dopo l'accorpamento:

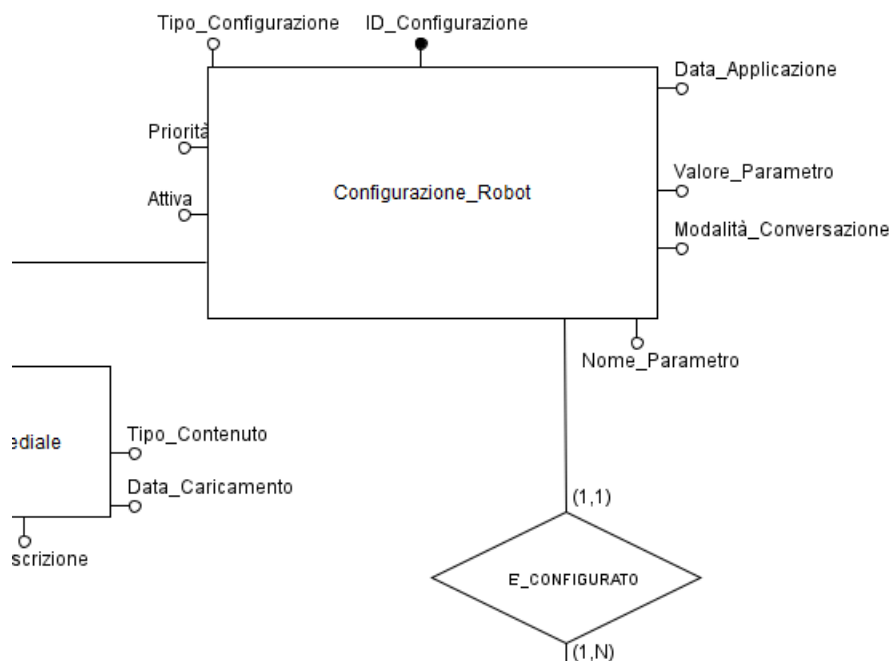


Accorpamento 3 (Competenza_Robot + Personalizzazione_Robot → Configurazione_Robot):

Porzione di schema prima dell'accorpamento:



Porzione di schema dopo l'accorpamento:



2.2. Altre ristrutturazioni:

Oltre agli accorpamenti, è stata effettuata un'altra ristrutturazione significativa per ottimizzare la chiarezza e l'efficienza del modello relazionale.

- **Eliminazione della Relazione partecipa tra utente e interazione:**
- **Motivazioni Dettagliate:** Nello schema E-R iniziale non ristrutturato, una relazione diretta 'partecipa' tra utente e interazione era stata inserita per esplicitare la connessione logica. Tuttavia, in fase di ristrutturazione, si è riconosciuto che questa relazione è già **implicitamente e più precisamente gestita dall'entità sessione**.
- **Ruolo di sessione come Intermediario:** L'entità sessione è il legame naturale tra utente e interazione. Un utente avvia una sessione, e quella sessione contiene una o più interazioni. Quindi, l'interazione è già intrinsecamente collegata all'utente tramite la sessione di cui fa parte.
- **Riduzione della Ridondanza Relazionale:** Mantenere una relazione diretta utente-interazione avrebbe creato un percorso ridondante e meno preciso per collegare un'interazione al suo utente, aumentando la complessità dello schema senza aggiungere valore informativo. L'eliminazione ha portato a un modello più pulito e conforme ai principi di normalizzazione, dove ogni relazione ha un significato univoco e non duplicato.

2.3. Scelta degli identificatori principali

Entità	Identificatore principale	Motivazioni della scelta
Utente	ID_Utente	L'email/nome/cognome potrebbe cambiare o duplicarsi (es. re-iscrizione). Un ID numerico è stabile e più efficiente.
Accessibilità	ID_Accessibilita	Attributi che possono variare e presentare una struttura interna complessa.
Profilo_Utente	ID_Profilo	Profilo dinamico e personalizzato. ID surrogato è necessario, inoltre è coerente con la relazione 1:N tra

		profilo_utente ed Utente.
Interfaccia_Canale	ID_Interfaccia	Identificativo unico per ogni canale e configurazione, anche se presentassero lo stesso nome o tipo.
Config_Privacy	ID_Config	Configurazioni distinte, facilmente tracciabili per auditing o GDPR.
Sessione	ID_Sessione	Token non è affidabile come chiave, ha uno scopo cruciale ma diverso da quello di chiave primaria, la sua funzione è garantire sicurezza e l'autenticazione dell'utente durante la sessione. Inoltre un token ha una lunghezza variabile il che li rende meno efficienti, richiedono algoritmi specifici per garantire l'unicovità e la casualità e anche se un token è univoco per una data sessione, i token stessi sono spesso progettati per scadere o essere revocati per ragioni di sicurezza, una chiave primaria non dovrebbe mai avere queste caratteristiche.
Metrica	ID_Metrica	Una singola interazione può produrre molteplici metriche, un ID univoco per ogni metrica permette di registrare tutte queste misurazioni in modo separato e dettagliato.
Storico_Emotivo	ID_Storico	Raccoglie molte osservazioni nel tempo.

		Un ID stabile semplifica la gestione.
Feedback	ID_Feedback	Possono esserci più feedback per utente. Un ID surrogato permette modifiche, tracciabilità.
Interazione	ID_Interazione	Ogni scambio deve essere tracciabile e univoco. ID surrogato è ideale.
Robot	ID_Robot	Identificazione semplice e autonoma anche se il nome è uguale tra istanze. Serve per distinguere un robot da un altro anche se sono dello stesso modello e hanno lo stesso nome, questo è essenziale per il tracking, il debugging e la gestione.
Messaggio	ID_Messaggio	Unifica risposta e suggerimento. Serve per gestione, traduzioni e tracciamento.
Traduzione	ID_Traduzione	Possono esistere più traduzioni per lo stesso messaggio. Un ID facilita la gestione.
Policy_Moderazione	ID_Policy	Ogni regola è autonoma e può essere modificata. ID permette gestione sicura e flessibile.
Log_Errori	ID_Log	Più errori possono verificarsi, anche per la stessa sessione. ID sequenziale è perfetto.
Configurazione_Robot	ID_Configurazione	Le configurazioni possono essere molte e dinamiche: il surrogato garantisce unicità e prestazioni.

Contenuto_Multimediale	ID_Contentuto	Contenuti possono essere grandi file. L'ID evita duplicazioni e semplifica i riferimenti
Emozione	ID_Emozione	Diverse istanze di emozioni possono avere lo stesso nome, in quanto la rilevazione di una stessa emozione può ripetersi più volte in una singola sessione. Un ID surrogato garantisce l'unicità e la gestione di ogni singola istanza di emozione.

Per ogni entità del database, è stato scelto un **identificatore surrogato** (una chiave primaria artificiale) come identificatore principale. Questa è una decisione strategica e coerente per tutto il modello, basata su solidi principi di database.

Le chiavi naturali possono cambiare nel tempo. Se un dato che funge da chiave primaria cambia, diventerebbe un incubo aggiornare non solo la tabella principale ma anche tutte le tabelle correlate che utilizzano quella chiave come chiave esterna. Un identificatore surrogato, essendo un valore artificiale non cambia mai, questo garantisce stabilità e integrità referenziale nel database. Gli identificatori surrogati sono tipicamente numeri interi semplici. Questo li rende facili da usare, sono consistenti in quanto hanno un formato uniforme prevedibile a differenza delle chiavi naturali che possono avere formati diversi e sono indipendenti dal dominio.

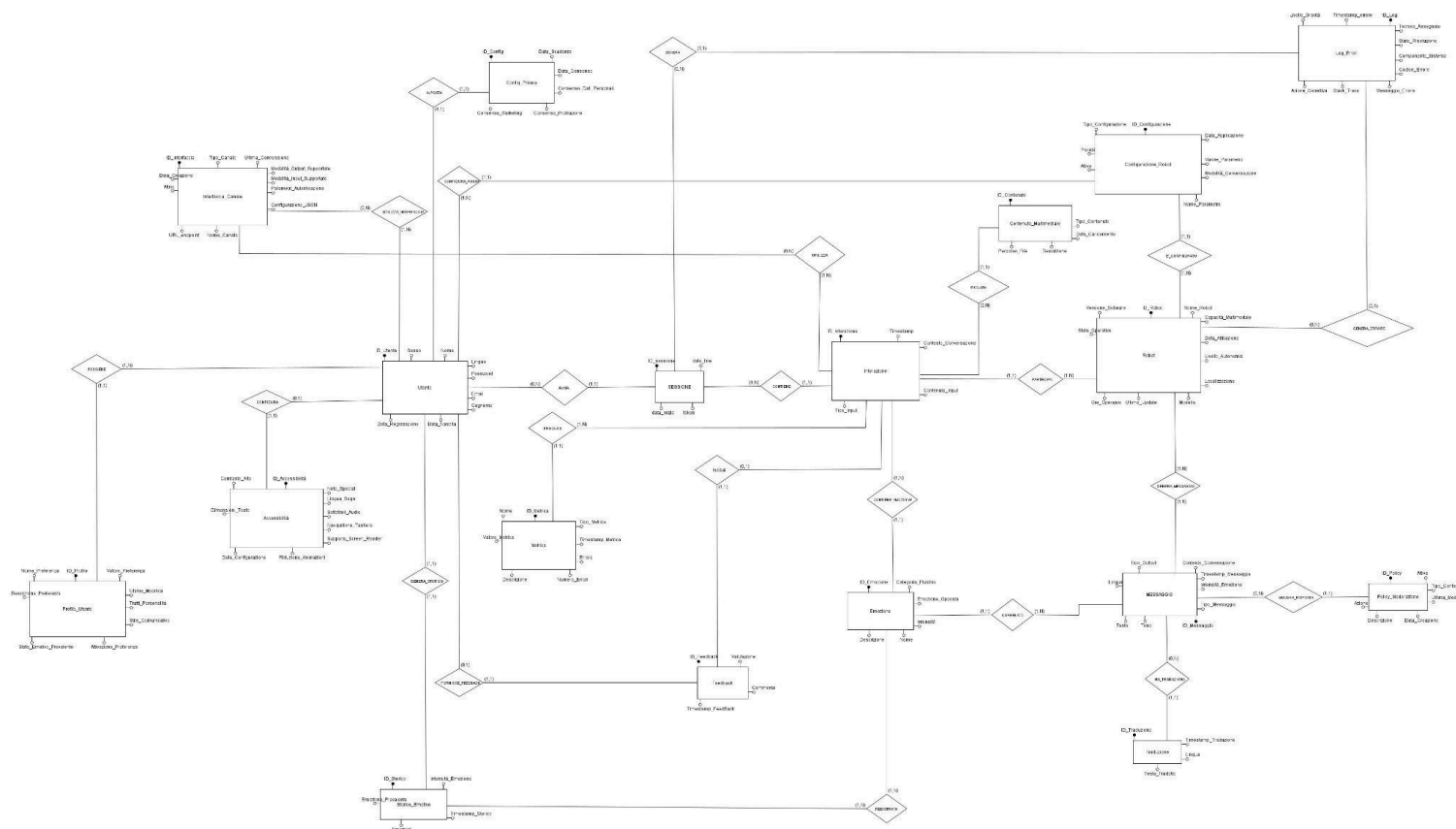
Inoltre gli ID surrogati essendo numeri interi offrono prestazioni migliori.

A volte, una chiave naturale potrebbe essere composta da molteplici attributi (chiave composta). Usare una chiave composta come chiave primaria e poi come chiave esterna in altre tabelle rende i join complessi. Un identificatore surrogato, essendo un valore singolo, semplifica notevolmente le relazioni tra tabelle.

Ulteriormente, generando un ID surrogato, il database si occupa automaticamente di garantire unicità alle chiavi primarie.

In alcuni contesti l'esposizione di chiavi naturali che contengono informazioni sensibili, può rappresentare un rischio per la privacy o la sicurezza. Gli identificatori surrogati, non avendo alcun significato al di fuori del database, offrono un livello di astrazione e protezione.

2.4 Schema E-R ristrutturato + regole



Regole aziendali

Vincoli di integrità:

RV1	Ogni profilo utente deve essere associato a uno e un solo utente, mentre un utente può possedere uno o più profili.
RV2	Ogni interazione deve essere associata a una sessione esistente.
RV3	Ogni interazione è associata ad almeno un'emozione registrata durante il suo svolgimento.
RV4	Ogni messaggio è associato a una o più emozioni, e ciascuna emozione rilevata nel contesto del messaggio può essere legata a un solo messaggio
RV5	Ogni messaggio deve essere generato da un solo robot.

RV6	Ogni feedback deve essere fornito da un solo utente e riferito a una sola interazione.
RV7	Ogni robot deve possedere almeno una configurazione.
RV8	Ogni contenuto multimediale deve essere associato a una sola interazione.
RV9	Ogni utente può fornire al massimo un solo feedback per ciascuna interazione, impedendo feedback duplicati.
RV10	Ogni traduzione deve riferirsi a un solo messaggio.
RV11	Ogni messaggio può essere associato a più traduzioni, ma ciascuna traduzione deve avere una lingua diversa.
RV12	Ogni configurazione robot deve essere riferita a un solo robot e, se creata da un utente, anche a un solo utente.
RV13	Ogni metrica deve essere collegata a una interazione e avere dati misurabili (es. tempo risposta > 0).
RV14	Ogni configurazione di accessibilità deve essere associata a un solo utente.
RV15	Ogni politica di moderazione è applicata a 1 o più messaggi
RV16	Ogni utente può avviare una sessione solo se correttamente autenticato.
RV17	Ogni sessione è avviata da un solo utente e può generare più interazioni.
RV18	Ogni interazione deve essere associata ad almeno un'interfaccia canale.
RV19	Ogni messaggio deve essere generato in risposta a un'emozione precedentemente rilevata.
RV20	Ogni messaggio deve essere coerente con lo stato emotivo rilevato durante l'interazione.
RV21	Ogni feedback deve essere riferito a una interazione conclusa
RV22	L'intensità delle emozioni deve essere compresa tra 0.0 e 1.0, secondo la scala di Plutchik.
RV23	Solo le emozioni primarie e secondarie di Plutchik possono essere registrate come nome nell'entità Emozione.

RV24	Emozioni opposte secondo Plutchik non devono coesistere nella stessa interazione.
RV25	Ogni log di errore deve essere associato a una sessione e/o a un robot.
RV26	Ogni contenuto multimediale deve essere classificato per tipo e deve avere un URL o riferimento valido.

Derivazioni:

RD1	Il tempo medio di risposta per utente si ottiene calcolando la media dei tempi nelle metriche associate alle sue interazioni.
RD2	Lo stato emotivo prevalente dell'utente si ottiene analizzando l'emozione più ricorrente nello Storico_Emotivo.
RD3	Il numero di interazioni per sessione si ottiene contando le interazioni associate a ciascuna sessione.
RD4	La lingua preferita dell'utente può essere derivata dalla lingua più frequente tra le traduzioni ricevute.
RD5	Il tasso di errore di un robot si ottiene dividendo il numero di log errori associati a quel robot per il numero di interazioni a cui ha partecipato.
RD6	Il tasso di feedback positivi si calcola come la percentuale di valutazioni superiori a una soglia (es. > 6) tra tutti i feedback ricevuti.
RD7	Il canale preferito dall'utente si ottiene contando la frequenza d'uso delle interfacce associate alle sue interazioni.
RD8	L'efficacia delle configurazioni del robot si valuta confrontando le metriche (tempo risposta, feedback) prima e dopo l'attivazione della configurazione.
RD9	La frequenza delle emozioni per utente si ottiene conteggiando il numero di volte in cui ogni emozione appare nello Storico_Emotivo.
RD10	Il tipo di messaggio da generare può essere derivato dalla combinazione di due emozioni primarie secondo la teoria di Plutchik.

RD11	Il numero medio di feedback per utente si ottiene dividendo il numero totale di feedback per il numero di utenti attivi.
RD12	Il livello di coinvolgimento emotivo di una interazione si calcola come media pesata dell'intensità delle emozioni associate.
RD13	Le performance del robot con configurazione attiva si ottengono confrontando i dati di metrica prima e dopo l'attivazione della configurazione.
RD14	Il tasso di moderazione attiva si ottiene dividendo il numero di messaggi soggetti a policy per il numero totale di messaggi generati.
RD15	Il livello di coinvolgimento emotivo di un utente si ottiene calcolando la media delle intensità nel suo Storico_Emotivo.

2.5 Schema relazionale con vincoli di integrità referenziale

Utente (id_utente, nome, cognome, email, password, sesso, data_nascita, lingua, data_registrazione)

Profilo_Utente (id_profilo, nome_preferenza, tratti_personalita, stato_emotivo_prevalente, attivazione_preferenza, descrizione_preferenza, valore_preferenza, stile_comunicativo, ultima_modifica, id_utente)

Profilo_Utente (id_utente) referencia Utente(id_utente)

Accessibilita (id_accessibilita, contrasto_alto, dimensioni_testo, data_configurazione, riduzione_animazioni, navigazione_tastiera, sottotitoli_audio, lingua_segna, note_speicali, supporto_screen_reader, id_utente)

Accessibilita (id_utente) referencia Utente(id_utente)

Config_Privacy (id_config, consenso_dati_personali, consenso_marketing, consenso_profilazione, data_consenso, data_scadenza, id_utente)

Config_Privacy (id_utente) referencia Utente(id_utente)

Sessione (id_sessione, token, data_inizio, data_fine, id_utente)

Sessione (id_utente) referencia Utente(id_utente)

Robot (id_robot, nome_robot, modello, versione_software, capacita_multimodale, data_attivazione, livello_autonomia, localizzazione, ultimo_update, ore_operative, stato_operativo)

Configurazione_Robot (id_configurazione, tipo_configurazione, nome_parametro, valore_parametro, modalita_conversazione, priorita, attiva, data_applicazione, id_robot, id_utente)

Configurazione_Robot (id_robot) referencia Robot(id_robot)

Configurazione_Robot (id_utente) referencia Utente(id_utente)

Interfaccia_Canale (id_interfaccia, tipo_canale, ultima_connessione, modalita_output_supportate, modalita_input_supportate, parametri_autenticazione, configurazione_json, nome_canale, url_endpoint, attivo, data_creazione)

Interazione (id_interazione, timestamp, contenuto_input, tipo_input, contesto_conversazione, id_sessione, id_robot)

Interazione (id_sessione) referencia Sessione(id_sessione)

Interazione (id_robot) referencia Robot(id_robot)

Utente_Interfaccia(id_utente, id_interfaccia)

Utente_interfaccia(id_utente) referencia Utente(id_utente)

Utente_Interfaccia(id_interfaccia) referencia Interfaccia_canale(id_interfaccia)

Interazione_Interfaccia (id_interazione, id_interfaccia)

Interazione_Interfaccia (id_interazione) referencia Interazione(id_interazione)

Interazione_Interfaccia (id_interfaccia) referencia Interfaccia_Canale(id_interfaccia)

Emozione (id_emozione, id_messaggio, id_interazione, nome, descrizione, intensità, categoria_plutchik, emozione_opposta)

Emozione(id_messaggio) referencia Messaggio(id_messaggio)

Emozione(id_interazione) referencia Interazione(id_interazione)

Storico_Emotivo (id_storico, id_utente, timestamp_storico, intensita_emozione, emozione_prevalente)

Storico_Emotivo (id_utente) referencia Utente(id_utente)

Storico_Emotivo_Emozione(id_storico, id_emozione)

Storico_Emotivo_Emozione(id_storico) referencia Storico_Emotivo(id_storico)

Storico_Emotivo_Emozione(id_emozione) referencia Emozione(id_emozione)

Messaggio (id_messaggio, testo, lingua, tono, tipo_messaggio, timestamp_messaggio, contesto_conversazione, intensita_emozione, tipo_output, id_robot)

Messaggio (id_robot) referencia Robot(id_robot)

Traduzione (id_traduzione, timestamp_traduzione, testo_tradotto, lingua, id_messaggio)

Traduzione (id_messaggio) referencia Messaggio(id_messaggio)

Feedback (id_feedback, valutazione, commento, timestamp_feedback, id_utente, id_interazione)

Feedback (id_utente) referencia Utente(id_utente)

Feedback (id_interazione) referencia Interazione(id_interazione)

Contenuto_Multimediale (id_contenuto, tipo_contenuto, data_caricamento, percorso_file, descrizione, id_interazione)

Contenuto_Multimediale (id_interazione) referencia Interazione(id_interazione)

Metrica (id_metrica, nome, valore_metrica, descrizione, timestamp_metrica, tipo_metrica, errore, numero_errori, id_interazione)

Metrica (id_interazione) referencia Interazione(id_interazione)

Log_Errori (id_log, codice_errore, livello_gravita, azione_correttiva, stack_trace, messaggio_errore, timestamp_errore, componente_sistema, stato_risoluzione, tecnico_assegnato, id_robot, id_sessione)

Log_Errori (id_robot) referencia Robot(id_robot)

Log_Errori (id_sessione) referencia Sessione(id_sessione)

Policy_Moderazione (id_policy, descrizione, azione, tipo_contenuto, attiva, data_creazione, ultima_modifica, id_messaggio)

Policy_Moderazione(id_messaggio) referencia Messaggio(id_messaggio)

Commenti sullo schema relazionale:

Il modello relazionale sviluppato rappresenta la traduzione fedele dello schema E-R concettuale ristrutturato, con particolare attenzione ai vincoli di integrità e una ristrutturazione dei dati coerente con i requisiti funzionali.

Ogni tabella ha una chiave primaria (id_%) per garantire l'unicità delle righe e facilitare le relazioni tra tabelle. Abbiamo scelto identificatori surrogati per evitare conflitti o duplicazioni, facilitando anche la scalabilità del DB.

Le chiavi esterne riflettono le relazioni tra entità nel modello concettuale, ad esempio la tabella 'Interazione' ha come chiave esterna id_utente, id_robot, mentre 'Feedback' ha id_interazione e id_utente come FK, assicurando la tracciabilità completa.

Lo schema logico è in terza forma normale (3FN) in quanto:

- 1NF: Tutti gli attributi sono atomici (non ci sono valori multipli in una colonna).
- 2NF: Nessuna dipendenza parziale da una chiave composta (le PK sono singole o composte correttamente).
- 3FN: Nessuna dipendenza transitiva da chiavi non primarie (ogni attributo dipende solo dalla chiave primaria).

Nello schema non ci sono attributi che dipendono da altri attributi non chiave o dati ripetuti inutilmente. Inoltre sono state introdotte tabelle ponte: 'interazione_interfaccia', 'utente_interfaccia' e 'storico_emotivo_emozione' per gestire le relazioni N:N tra: interazione-interfaccia_canale, utente-interfaccia_canale e storico_emotivo-emozione.

DDL E DML

1. DDL di creazione del database

Struttura generale:

Nel file 'Pietro-Mezzatesta-Ossama-Boulakhdar_DDL.sql' è contenuta la definizione completa delle tabelle del database progettato per supportare una piattaforma di Human-Robot Interaction con funzionalità di analisi emotiva e personalizzazione. La progettazione è basata su uno schema E-R esteso e riflette i vincoli concettuali emersi dall'analisi dei requisiti funzionali.

Il database è composto da 21 tabelle principali, organizzate in:

- Entità fondamentali: utente, robot, interfaccia_canale, messaggio, emozione, interazione, ecc...
- Tabelle di configurazione o profilo: profilo_utente, configurazione_robot, accessibilità, config_privacy.
- Tabelle di relazione (N:N): utente_interfaccia, interazione_interfaccia, storico_emotivo_emozione.
- Tabelle dinamiche/logiche: sessione, feedback, contenuto_multimediale, log_errori, metriche, storico_emotivo.

Vincoli di integrità referenziale:

Tutte le relazioni sono esplicitamente modellate con FOREIGN KEY, utilizzando:

- ON DELETE CASCADE: per garantire la propagazione automatica delle eliminazioni in tabelle indipendenti (esempio: utente viene eliminato, vengono eliminate preferenze, configurazioni, interazioni..).
- ON UPDATE CASCADE: per mantenere consistenza di aggiornamenti sugli identificatori (anche se rari).

Esempio:

```
FOREIGN KEY(id_utente) REFERENCES utente(id_utente)
```

```
ON DELETE CASCADE
```

```
ON UPDATE CASCADE
```

Chiavi primarie:

Ogni entità ha una chiave primaria esplicita autonumerata (SERIAL) per l'identificazione univoca, e ogni tabella ponte ha una chiave primaria composta, esempio:

PRIMARY KEY(id_interazione, id_interfaccia)

Questa scelta supporta interrogazioni efficienti e integrità logica.

Vincoli di controllo (CHECK):

Sono stati definiti numerosi vincoli CHECK per diversi attributi per garantire che i valori rientrino negli insiemi consentiti e ridurre errori applicativi.

Esempi:

Robot:

- Livello autonomia: (0 - nessuna autonomia, 1 - supporto umano, 2 - assistenza parziale, 3 - autonomia condizionata, 4 - alta autonomia, 5 - autonomia completa).
- Stato operativo: (On, Off, Standby, Paused, Error, Emergenza, Manutenzione).

Messaggio:

- tono: (Empatico, Rassicurante, Incoraggiante, Informativo, Persuasivo, Rispettoso).
- tipo_messaggio: (risposta, suggerimento).
- tipo_output: (audio, visivo, testuale, aptico, olfattivo, propriocettivo, multimodale).

Emozione:

- categoria_plutchik: (emozione primaria, emozione secondaria).
- intensità: deve essere compresa tra 0.00 e 1.00.

Interfaccia:

- tipo_canale, modalità_input_supportate, modalità_output_supportate, parametri_autenticazione vincolanti a set predefiniti (es. visivo, vocale, tattile, token, otp, ecc...).

Utente:

- sesso: (M, F, O).
- password: con lunghezza minima ≥ 8 .

Vincoli temporali e logici:

Abbiamo incluso vincoli temporali per assicurare coerenza cronologica.

Ad esempio la tabella sessione include un vincolo logico per garantire che `data_fine > data_inizio`.

Alcune entità (es. `feedback`, `storico_emotivo`) usano timestamp automatici o a cura dell'applicazione.

Tabelle ponte (N:N):

Le relazioni molti a molti sono implementate tramite tabelle intermedie con chiavi esterne e PRIMARY KEY composta:

- `utente_interfaccia(id_utente, id_interfaccia)`
- `interazione_interfaccia(id_interazione, id_interfaccia)`
- `storico_emotivo_emozione(id_storico, id_emozione)`

Questo garantisce univocità e permette interrogazioni efficienti.

Attributi multivalore e configurazioni:

Le configurazioni (sia utente che robot) sono modellate come tabelle separate per:

- rappresentare opzioni multiple e modificabili,
- mantenere la tracciabilità (`data_modifica`, `attiva`, `priorità`),
- gestire personalizzazioni complesse in modo scalabile.

Considerazioni conclusive:

L'intero schema è stato progettato per:

- Riflettere fedelmente la struttura concettuale emersa dallo studio del dominio HCI ed emotivo.
- Garantire coerenza, estensibilità, robustezza e integrità dei dati a fronte di operazioni frequenti di inserimento, aggiornamento e cancellazione.
- Supportare in modo scalabile query analitiche su emozioni, metriche, preferenze e feedback.

2. DML di popolamento di tutte le tabelle del database

Attraverso il DML di popolamento abbiamo cercato di simulare scenari realistici, che si potrebbero verificare all'interno di un sistema di Human-Robot Interaction (HRI). L'obiettivo era quello di inserire dati coerenti e significativi per ogni tabella, in modo da rendere possibile l'esecuzione e la validazione di query complesse, join, vincoli di integrità e derivazioni previste dal progetto.

Le tabelle principali come Utente e Robot sono state popolate con almeno 10 istanze ciascuna, per rappresentare una varietà di profili e casi d'uso. Ad esempio, ogni utente ha caratteristiche uniche come nome, cognome, email, sesso, lingua preferita e data di nascita, mentre i robot sono diversificati per modello, versione software, capacità multimodale e stato operativo. Questo consente di simulare un ambiente dinamico, con utenti e agenti artificiali dotati di comportamenti e configurazioni differenti.

Le restanti entità — come Interazione, Feedback, Configurazione_Privacy, Accessibilità, Contenuto_Multimediale, Traduzione, Messaggio, Emozione — sono state invece popolate con circa 5 istanze ciascuna, bilanciando la copertura dei casi con l'efficienza. L'obiettivo non era riempire il database con grandi moli di dati, ma garantire un set minimo sufficiente per testare tutti i vincoli, le relazioni (inclusi i casi N:N), e le funzionalità chiave dell'applicazione.

Inoltre, sono stati rispettati tutti i vincoli di integrità referenziale: ogni interazione è legata a una sessione valida, ogni emozione a una interazione o un messaggio, ogni configurazione robot a un robot e a un utente, ogni feedback a una specifica coppia utente-interazione, e così via. Anche le relazioni complesse, come Storico_Emotivo_Emozione o Interazione_Interfaccia, sono state popolate con attenzione, assicurandosi che i dati risultassero coerenti con la semantica del dominio.

Questa fase di popolamento ha svolto quindi una duplice funzione: da un lato, garantire la coerenza strutturale del modello, dall'altro offrire un terreno pratico per testare interrogazioni, aggiornamenti e controlli logici sul funzionamento complessivo del sistema. Grazie a ciò, il database è ora pronto a supportare test avanzati, operazioni di lettura e scrittura frequenti, analisi delle performance e simulazioni reali.

3. DML di modifica

Le operazioni di modifica rappresentano scenari reali e frequenti nell'uso quotidiano del sistema Human-Robot Interaction. Sono state progettate per allinearsi ai requisiti funzionali esplicitati e per rispettare le esigenze ricche di aggiornamento e gestione delle informazioni in un database relazionale.

1. Inserimento di un nuovo utente

Questa è una delle operazioni più frequenti, necessaria ogni volta che un nuovo individuo accede al sistema per la prima volta. L'inserimento del record nella tabella utente abilita tutte le funzionalità del sistema legate a sessioni, feedback, interazioni, ecc.

Questa operazione corrisponde all'avvio del rapporto tra l'utente e il sistema. L'utente fornisce i propri dati anagrafici e le informazioni necessarie per essere identificato e autenticato. È la base di qualsiasi altro comportamento del sistema ed è pertanto tra le operazioni più frequenti e critiche.

- Frequenza: alta
- Operazione collegata: Op1 – Registrazione utente con profilo e preferenze
- Entità coinvolte: Utente

2. Aggiunta di un feedback a un'interazione

Il feedback rappresenta l'opinione esplicita dell'utente rispetto a un'interazione appena conclusa. È fondamentale per l'auto-miglioramento del sistema, il monitoraggio della qualità del servizio e l'adattamento ai bisogni dell'utente.

- Frequenza: elevata dopo ogni interazione
- Operazione collegata: Op3 – Rilevazione emozioni e feedback
- Entità coinvolte: Feedback, Utente, Interazione

3. Modifica dello stato operativo di un robot

Il robot può cambiare stato più volte nel suo ciclo di vita (ad es. da "On" a "Manutenzione"). Questi aggiornamenti sono fondamentali per indicare la disponibilità o meno del sistema all'interazione.

- Frequenza: media (periodica o in risposta a eventi tecnici)
- Operazione collegata: Op8 – Archiviazione e gestione delle interazioni
- Entità coinvolte: Robot

4. Attivazione di una configurazione robot

Quando un utente sceglie di personalizzare il comportamento del robot, l'attivazione di una configurazione consente di adattare l'esperienza alle proprie esigenze. Questo include ad esempio il tono della voce, la modalità empatica o le risposte preferite.

- Frequenza: elevata dopo la creazione di nuove configurazioni
- Operazione collegata: Op7 – Personalizzazione robot
- Entità coinvolte: Configurazione_Robot, Utente, Robot

5. Cambio della lingua preferita di un utente

Gli utenti possono modificare le preferenze linguistiche in qualsiasi momento, rendendo l'interfaccia e le risposte più comprensibili. Questa operazione è tipica nei contesti multilingue.

- Frequenza: media
- Operazione collegata: Op12 – Supporto multilingua
- Entità coinvolte: Utente

6. Eliminazione di un contenuto multimediale obsoleto

Nel tempo, i contenuti associati alle interazioni potrebbero non essere più utili o rilevanti. La loro eliminazione consente di mantenere il database leggero e aggiornato.

- Frequenza: periodica
- Operazione collegata: Op8 – Archiviazione e ricerca interazioni
- Entità coinvolte: Contenuto_Multimediale

7. Aggiornamento del tono di un messaggio robot

Per migliorare la qualità dell'esperienza utente, si può decidere di aggiornare il tono usato nei messaggi. Questo intervento riflette esigenze di stile, coerenza o correzione.

- Frequenza: media
- Operazione collegata: Op4 – Generazione risposte basate su emozioni
- Entità coinvolte: Messaggio, Robot

8. Rimozione di un feedback errato

Un utente potrebbe richiedere la cancellazione di un feedback fornito erroneamente. Ciò garantisce l'affidabilità dei dati raccolti.

- Frequenza: bassa ma necessaria
- Operazione collegata: Op9 – Apprendimento da feedback
- Entità coinvolte: Feedback

9. Aggiornamento timestamp ultima connessione interfaccia

Ogni volta che un'interfaccia viene utilizzata per avviare un'interazione, si aggiorna la data/ora dell'ultima connessione. Questo aiuta a monitorare l'utilizzo delle interfacce.

- Frequenza: alta, automatica ad ogni utilizzo dell'interfaccia
- Operazione collegata: Op2 – Interazione utente-robot tramite interfaccia
- Entità coinvolte: Interfaccia_Canale

10. Inserimento di una nuova configurazione privacy

Al primo accesso, ogni utente deve fornire i propri consensi sulla privacy. Questa operazione è obbligatoria e quindi molto comune.

- Frequenza: una tantum per ogni nuovo utente
- Operazione collegata: Op11 – Gestione privacy e sicurezza
- Entità coinvolte: Configurazione_Privacy, Utente

11. Attivazione di sottotitoli e screen reader

Utile per utenti con disabilità visive o uditive. Questa modifica rende il sistema inclusivo e accessibile.

- Frequenza: medio-alta, dipende dai bisogni dell'utente
- Operazione collegata: Op11 – Gestione accessibilità e sicurezza
- Entità coinvolte: Accessibilità, Utente

12. Registrazione di una nuova emozione rilevata

Le emozioni rilevate sono centrali nella personalizzazione. Questa operazione avviene ogni volta che viene effettuata un'analisi emotiva

- Frequenza: alta, dopo ogni interazione significativa
- Operazione collegata: Op3 – Rilevazione emozioni
- Entità coinvolte: Emozione, Interazione, Messaggio

13. Inserimento di una traduzione in una nuova lingua

Serve per estendere il sistema a più lingue. Può essere attivata al momento dell'invio del messaggio o successivamente.

- Frequenza: media
- Operazione collegata: Op5 – Traduzione risposte
- Entità coinvolte: Traduzione, Messaggio, Utente

14. Modifica del tono del messaggio

Come per l'operazione 7, permette di ritardare la comunicazione del robot in funzione della situazione specifica.

- Frequenza: media
- Operazione collegata: Op6 – Adattamento comunicativo
- Entità coinvolte: Messaggio

15. Eliminazione di una sessione scaduta

Per motivi di sicurezza e per ottimizzare lo spazio, si eliminano periodicamente sessioni non più attive.

- Frequenza: automatica o periodica
- Operazione collegata: Op11 – Gestione sessioni e sicurezza
- Entità coinvolte: Sessione

16. Correzione della data di ultima modifica nel profilo utente

Quando un utente aggiorna le sue preferenze o lo stile comunicativo, è importante registrare il momento della modifica.

- Frequenza: media
- Operazione collegata: Op1 – Gestione profilo utente
- Entità coinvolte: Profilo_Utente

17. Inserimento di una metrica relativa a un'interazione

Questa operazione è essenziale per tracciare e analizzare le prestazioni del sistema durante un'interazione. Le metriche possono includere, ad esempio, il tempo di risposta del robot, il numero di errori rilevati o altri indicatori utili a valutare l'efficacia e la fluidità della conversazione. Tali informazioni sono cruciali per il monitoraggio, l'ottimizzazione e l'adattamento dinamico del comportamento del

sistema intelligente. L'inserimento di una nuova metrica avviene di norma subito dopo o durante un'interazione.

- Frequenza: alta (una per ogni interazione)
- Operazione collegata: Op8 – Archiviazione e monitoraggio delle interazioni
- Entità coinvolte: Metrica, Interazione, Robot

18. Inserimento di un log di errore

Questa operazione consente di registrare eventuali malfunzionamenti o anomalie nel comportamento del robot o della sessione. È essenziale per il tracciamento e il debug.

- Frequenza: media
- Operazione collegata: Op11 – Gestione sicurezza
- Entità coinvolte: Sessione, Robot, Log_errori

19. Modifica della priorità di una configurazione

Serve per adattare dinamicamente il comportamento del robot in base alla situazione o alle preferenze dell'utente.

- Frequenza: media-alta
- Operazione collegata: Op6 – Competenze del robot, comprensione input, adattamento allo stile dell'utente.
- Entità coinvolte: Configurazione_Robot

20. Aggiornamento emozione prevalente nello storico

Quando un'analisi post-interazione aggiorna lo stato emotivo dominante, utile per l'apprendimento automatico.

- Frequenza: alta
- Operazione collegata: Op4 – Generazione di risposte e suggerimenti basati su emozione e contesto.
- Entità coinvolte: Storico_Emotivo

21. Correzione del parametro di configurazione robot

Quando un parametro applicato non è adeguato e va modificato.

- Frequenza: media
- Operazione collegata: Op7 – Personalizzazione del robot da parte dell'utente.
- Entità coinvolte: Configurazione_Robot

22. Aggiornamento parametri autenticazione interfaccia

Quando si decide di usare un metodo più sicuro (es. passaggio da password a MFA).

- Frequenza: media-bassa
- Operazione collegata: Op11 – Gestione delle sessioni, autenticazione e sicurezza.
- Entità coinvolte: Interfaccia_Canale