

# Analisi dei Requisiti

versione 1.0.0



7commits@gmail.com

Progetto di Ingegneria del Software

A.A. 2024/2025

Destinatari	Responsabile	Redattori	Verificatori
Prof. Tullio Vardanega	Marco Cola	Stefano Dal Poz	Giulia Hu
Prof. Riccardo Cardin		Marco Cola	Ruize Lin
Gruppo 🇮🇹 7Commits		Ruize Lin	Michele Ogniben
		Mattia Piva	Giada Rossi
		Giada Rossi	Marco Cola

## Registro delle modifiche

Versione	Data	Autori	Verificatori	Descrizione
v1.0.0	2025-06-12	Marco Cola	Michele Ogniben	Approvazione per colloquio RTB
v0.2.6	2025-06-11	Giulia Hu	Marco Cola, Giada Rossi	Corrette immagini UML modificati
v0.2.5	2025-06-10	Giada Rossi	Marco Cola, Giulia Hu, Ruize Lin	Aggiunti Casi d'Uso UC7, UC12, UC14, UC16, UC19 e UC25 in seguito a revisione interna e rinumerazione Casi d'Uso di conseguenza
v0.2.4	2025-05-17	Marco Cola	Marco Cola, Giada Rossi, Stefano Dal Poz	Corrette tabelle dei requisiti
v0.2.3	2025-05-15	Stefano Dal Poz	Marco Cola, Giada Rossi, Ruize Lin	Aggiunta priorità nella notazione dei requisiti
v0.2.2	2025-05-15	Giulia Hu, Marco Cola	Marco Cola, Giada Rossi, Ruize Lin	Aggiunte immagini corrette UML Casi d'Uso
v0.2.1	2025-05-10	Marco Cola, Ruize Lin, Mattia Piva	Marco Cola, Giada Rossi	Aggiunti Casi d'Uso UC22: «Salvataggio risultato test» e UC21: «Visualizzazione risultati test»
v0.2.0	2025-05-06	Marco Cola, Ruize Lin	Marco Cola, Giada Rossi, Ruize Lin	Revisione casi d'uso in seguito a colloquio con proponente
v0.1.4	2025-04-28	Marco Cola	Giada Rossi, Michele Ogniben, Ruize Lin, Marco Cola	Corrette tabelle sezioni 4) e 5)
v0.1.3	2025-04-26	Marco Cola, Giulia Hu	Giada Rossi, Michele Ogniben, Ruize Lin, Marco Cola	Rimosso UC6 per ridondanza, Aggiunte immagini dei Casi d'Uso da UC1 a UC21
v0.1.2	2025-04-16	Marco Cola	Giada Rossi, Michele Ogniben, Ruize Lin, Marco Cola	Rinumerati UC in seguito all'eliminazione di UC8, rimossi UC26 e UC27 per ridondanza
v0.1.1	2025-04-15	Ruize Lin	Giada Rossi, Michele Ogniben, Ruize Lin, Marco Cola	Corretti casi d'uso UC4, UC8, UC16, Aggiunti UC26, UC27
v0.1.0	2025-04-15	Marco Cola	Giulia Hu, Giada Rossi, Michele	Aggiunti link al Glossario

			Ogniben, Ruize Lin, Marco Cola	
v0.0.9	2025-04-14	Marco Cola	Giulia Hu, Giada Rossi, Michele Ogniben, Ruize Lin, Marco Cola	Iniziato a riempire tabelle sezione 4: 4.1, 4.2, 4.3
v0.0.8	2025-04-14	Marco Cola, Ruize Lin	Michele Ogniben, Ruize Lin, Marco Cola	Aggiunti UC18, UC18.1, UC19, UC19.1, UC19.2, UC20, UC21, UC21.1, UC22, UC23, UC24, UC25
v0.0.7	2025-04-13	Marco Cola, Ruize Lin	Michele Ogniben, Ruize Lin, Marco Cola	Aggiunti UC7, UC8, UC9, UC10, UC11, UC12, UC13, UC13.1, UC14, UC15, UC16, UC17
v0.0.6	2025-04-13	Marco Cola	Michele Ogniben, Ruize Lin, Marco Cola	Aggiunte generalizzazioni di UC1: UC1.1, 1.2, 1.3, 1.4, 1.5 in quanto relativi alle domande
v0.0.5	2025-04-13	Marco Cola	Ruize Lin, Marco Cola	Aggiunta Sezione 5.2 e tabelle
v0.0.4	2025-04-11	Mattia Piva	Ruize Lin, Marco Cola	Aggiunti Casi d'Uso UC1, UC2, UC3, UC4, UC5
v0.0.3	2025-04-09	Marco Cola	Ruize Lin, Marco Cola	Aggiunte sezioni da 3.1 a 3.3 dei Casi d'Uso
v0.0.2	2025-04-09	Marco Cola	Ruize Lin, Marco Cola	Aggiunta 1) Introduzione e 2) Descrizione generale del Prodotto
v0.0.1	2025-04-08	Marco Cola	Ruize Lin, Marco Cola	Prima Bozza

## Indice

1. <b>Introduzione</b> .....	6
1.1. Scopo del documento .....	6
1.2. Glossario .....	6
1.3. Riferimenti .....	7
1.3.1. Riferimenti normativi .....	7
1.3.2. Riferimenti informativi .....	7
2. <b>Descrizione generale del prodotto richiesto da Zucchetti</b> .....	8
2.1. Obiettivo fissato .....	8
2.2. Funzionalità del prodotto e requisiti .....	8
2.2.1. <b>Caratteristiche e Requisiti obbligatori:</b> .....	8
2.2.2. <b>Requisiti opzionali suggeriti:</b> .....	9
2.3. Caratteristiche utente .....	9
2.3.1. Utenti del nostro prodotto .....	9
2.4. Tecnologie ed Analisi della struttura di progetto .....	10
3. <b>Casi d'uso</b> .....	11
3.1. Obiettivi .....	11
3.2. Introduzione ai Casi d'Uso .....	11
3.3. Attori .....	11
3.4. Elenco dei Casi d'Uso .....	12
3.4.1. UC: Gestione domande e risposte attese .....	12
3.4.1.1. UC1: Visualizzazione lista domande .....	12
3.4.1.1.1. UC1.1: Visualizzazione singola domanda .....	12
3.4.1.2. UC2: Annullamento eliminazione domanda .....	15
3.4.1.3. UC3: Filtro delle domande per Categoria di appartenenza .....	15
3.4.1.4. UC4: Aggiunta manuale domanda .....	16
3.4.1.5. UC5: Errore «Sono necessarie sia la domanda che la risposta attesa» .....	16
3.4.1.6. UC6: Aggiunta da file strutturato .....	17
3.4.1.7. UC7: Errore «Il file importato deve contenere le colonne “domanda” e “risposta_attesa”» .....	17
3.4.2. UC: Gestione dei set di domande .....	18
3.4.2.1. UC8: Visualizzazione lista dei set .....	18
3.4.2.1.1. UC8.1: Visualizzazione singolo set .....	18
3.4.2.2. UC9: Annullamento eliminazione set .....	20
3.4.2.3. UC10: Filtro dei set per Categoria di appartenenza .....	21
3.4.2.4. UC11: Aggiunta manuale set .....	21
3.4.2.5. UC12: Errore «Il nome del set è obbligatorio.» .....	22
3.4.2.6. UC13: Aggiunta set da file strutturato .....	22
3.4.2.7. UC14: Errore «Formato JSON non valido.» .....	22
3.4.3. UC: Configurazione API LLM .....	23
3.4.3.1. UC15: Creazione di Preset per la Configurazione API LLM .....	23
3.4.3.1.1. UC15.1: Test di connessione API .....	23
3.4.3.2. UC16: Errore «Il nome del preset non può essere vuoto.» .....	24
3.4.3.3. UC17: Visualizzazione lista dei preset configurati .....	24
3.4.3.3.1. UC17.1: Modifica preset .....	25
3.4.3.3.2. UC17.2: Elimina preset .....	25
3.4.4. UC: Avvio Test .....	26
3.4.4.1. UC18: Esecuzione test .....	26
3.4.4.1.1. UC18.1: Visualizzazione preset di configurazione API salvati . . .	27

3.4.4.1.2.	UC18.2: Visualizzazione set di domande salvate .....	27
3.4.4.2.	UC19: Errore «Incorretta chiave API fornita» .....	28
3.4.4.3.	UC20: Valutazione automatica .....	28
3.4.5.	UC: Visualizzazione dei risultati .....	29
3.4.5.1.	UC21: Visualizzazione risultati .....	29
3.4.5.1.1.	UC21.1: Visualizzazione risultati per singola domanda .....	30
3.4.5.1.2.	UC21.2: Visualizzazione metrica per singola risposta .....	30
3.4.5.2.	UC22: Salvataggio risultati del test .....	31
3.4.5.3.	UC23: Download risultati del test .....	32
3.4.5.4.	UC24: Caricamento risultati da file .....	32
3.4.5.5.	UC25: Errore «Caricamento risultati da file non riuscito» .....	33
3.4.5.6.	UC26: Visualizzazione storico dei test .....	33
3.4.5.6.1.	UC26.1: Visualizzazione elemento esecuzione test .....	34
3.4.5.7.	UC27: Applicazione filtri storico dei test .....	34
3.4.5.7.1.	UC27.1: Filtro storico dei test per nome del set .....	35
3.4.5.7.2.	UC27.2: Filtro storico dei test per LLM .....	35
3.4.5.7.3.	UC27.3: Filtro storico dei test per risultati di valutazione .....	36
3.4.5.8.	UC28: Confronto tra esecuzioni diverse di test .....	36
3.4.5.9.	UC29: Visualizzazione risultato confronto esecuzioni .....	37
3.4.5.9.1.	UC29.1: Visualizzazione elemento confronto esecuzione .....	38
3.4.5.9.2.	UC29.2: Visualizzazione confronto esecuzione singola domanda .	39
4.	<b>Requisiti</b> .....	40
4.1.	Requisiti di Funzionalità .....	40
4.2.	Requisiti di Qualità .....	44
4.3.	Requisiti di Vincolo .....	45
5.	<b>Tracciamento dei Requisiti</b> .....	46
5.1.	Riepilogo .....	46
5.2.	Conclusioni .....	47

# 1. Introduzione

## 1.1. Scopo del documento

L'*Analisi dei Requisiti<sub>G</sub>* è un documento essenziale in ogni progetto di sviluppo software che intenda aderire agli standard di qualità previsti dalla disciplina dell'Ingegneria del Software. Il suo scopo principale è definire in modo chiaro e strutturato le *funzionalità<sub>G</sub>* che il sistema dovrà offrire, ossia i *requisiti<sub>G</sub>* obbligatori, desiderati e opzionali che il prodotto dovrà soddisfare per risultare conforme alle esigenze espresse dal *proponente<sub>G</sub>*.

L'analisi non deve proporre direttamente una soluzione al problema, ma deve tener conto della fattibilità tecnologica, tracciando così un confine netto tra la definizione del problema e la sua progettazione (design).

In particolare, le finalità di questo documento possono essere sintetizzate nei seguenti punti:

- **Esplicitare le esigenze del proponente:** il documento raccoglie e formalizza le aspettative che il proponente ha nei confronti del prodotto software, includendo sia i requisiti esplicitamente indicati nel *capitolato<sub>G</sub>* sia quelli impliciti emersi durante le interviste e l'analisi.
- **Ridurre le ambiguità comunicative:** fornisce un punto di riferimento condiviso tra il team di sviluppo e il proponente, assicurando una comprensione univoca dei requisiti e delle funzionalità attese.
- **Garantire il tracciamento dei requisiti:** consente di mappare le esigenze del proponente attraverso l'identificazione, classificazione e organizzazione dei *casi d'uso<sub>G</sub>* e dei relativi requisiti, suddivisi in funzionali, qualitativi e di vincolo.
- **Fornire una base solida per la progettazione del sistema:** il documento rappresenta un input fondamentale per l'attività di progettazione, in quanto definisce cosa il sistema deve fare, senza però specificare come. Questo permette al team di identificare soluzioni tecnicamente adeguate.
- **Verifica e Validazione dei requisiti:** la *verifica dei requisiti<sub>G</sub>* mira a garantire che i requisiti siano formalmente corretti, completi e privi di contraddizioni. La *validazione dei requisiti<sub>G</sub>* invece, si concentra sull'allineamento tra quanto definito nei requisiti e le reali esigenze del proponente, assicurandosi che il prodotto finale rispecchi le aspettative.
- **Minimizzare i rischi e ottimizzare i costi di sviluppo:** una definizione accurata e condivisa dei requisiti riduce il rischio di errori e ritardi nelle fasi successive, contribuendo a contenere i costi e migliorare l'efficienza del progetto.
- **Formalizzare graficamente i requisiti:** al termine dell'analisi, i risultati vengono rappresentati in maniera formale tramite diagrammi dei casi d'uso, utili per visualizzare le interazioni tra gli *attori<sub>G</sub>* e le funzionalità del *sistema software<sub>G</sub>*.

## 1.2. Glossario

Per evitare ambiguità o fraintendimenti legati alla terminologia utilizzata nel documento, si è deciso di introdurre un glossario contenente le definizioni dei termini specifici del dominio d'uso. In questo modo, tutti i concetti rilevanti verranno chiariti e resi accessibili. I termini inclusi nel glossario sono contrassegnati nel testo con una G a pedice, in *questo modo<sub>G</sub>*, sono anche presenti nel documento *Glossario\_v1.0.0* disponibile nel sito web del gruppo  7Commits.

## 1.3. Riferimenti

### 1.3.1. Riferimenti normativi

- Norme\_di\_Progetto\_v1.0.0
- Capitolato C1: **Artificial QI** - Valutare i risultati di sistemi che utilizzano i Large Language Models.  
<https://www.math.unipd.it/~tullio/IS-1/2024/Progetto/C1.pdf>

### 1.3.2. Riferimenti informativi

- T5-Analisi dei requisiti:  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T05.pdf>
- P2-Diagrammi dei Casi d'Uso (UML):  
<https://www.math.unipd.it/~rcardin/swea/2022/Diagrammi%20Use%20Case.pdf>
- Large Language Models: A Survey:  
<https://arxiv.org/pdf/2402.06196>
- Retrieval-Augmented Generation for Large Language Models: A Survey  
<https://arxiv.org/pdf/2312.10997>
- Glossario\_v1.0.0:  
[https://7commits.github.io/7Commits/docs/RTB/Interni/Glossario\\_v1.0.0.pdf](https://7commits.github.io/7Commits/docs/RTB/Interni/Glossario_v1.0.0.pdf)

## 2. Descrizione generale del prodotto richiesto da Zucchetti

Il progetto, proposto da Zucchetti S.p.A., prevede la realizzazione di una piattaforma *software<sub>G</sub>* denominata **Artificial QI**, concepita per valutare le prestazioni di sistemi basati su *Large Language Models (LLM)<sub>G</sub>*. L'obiettivo è quello di offrire uno strumento integrato che consenta di confrontare, in modo efficace e automatizzato, le risposte fornite da un LLM con quelle attese per una determinata serie di domande.

La piattaforma dovrà permettere all'utente di:

- inserire, modificare e gestire coppie di domande e risposte attese tramite un'*interfaccia<sub>G</sub> user-friendly<sub>G</sub>* ;
- selezionare un sistema LLM da testare (tramite chiamate a *API<sub>G</sub>* conformi allo standard *OpenAPI 3.1<sub>G</sub>*) e sottoporli le domande archiviate;
- valutare automaticamente la coerenza tra le risposte ricevute dal modello e le risposte attese, tramite metodi sia tradizionali (es. *BM25<sub>G</sub>*) sia basati su altri LLM;
- visualizzare i risultati delle esecuzioni dei *test<sub>G</sub>* attraverso una sintesi numerica e una presentazione dei casi migliori e peggiori;
- eventualmente archiviare i risultati per confrontare performance tra diverse esecuzioni e modelli.

L'intero sistema deve essere **organico** e **coerente**, evitando la separazione in strumenti indipendenti. Tutte le funzionalità richieste devono coesistere all'interno della stessa applicazione, al fine di supportare un flusso di lavoro continuo ed efficiente per lo sviluppatore.

### 2.1. Obiettivo fissato

L'obiettivo del progetto consiste nello sviluppo di una piattaforma che permetta:

- di testare e confrontare in modo sistematico le risposte generate da LLM su una varietà di domande significative;
- di fornire un ambiente di validazione oggettiva per sviluppatori, consentendo loro di analizzare l'impatto delle modifiche al sistema in fase di sviluppo;
- di disporre di uno strumento affidabile e riutilizzabile per valutare miglioramenti, regressioni o effetti collaterali nel comportamento di un modello linguistico rispetto alle richieste dell'utente.

Il prodotto finale dovrà essere realizzato in più fasi:

- **Proof of Concept (*PoC<sub>G</sub>*)**: focalizzata sull'esplorazione delle tecnologie necessarie allo sviluppo.
- **Minimum Viable Product (*MVP<sub>G</sub>*)**: realizzazione completa delle funzionalità obbligatorie.
- **Prodotto finale**: estensione mediante requisiti opzionali e ottimizzazione generale.

### 2.2. Funzionalità del prodotto e requisiti

#### 2.2.1. Caratteristiche e Requisiti obbligatori:

1. **Archiviazione di una lista di domande e delle risposte attese**: sistema di raccolta persistente (*database<sub>G</sub>*) accessibile tramite interfaccia grafica per gestione *CRUD<sub>G</sub>*.
2. **Programma di esecuzione del test che attraverso una API ponga le domande ad un programma esterno e ne registri la risposta**: sistema automatizzato di *test<sub>G</sub>* che sottopone le domande a un LLM via *API Rest<sub>G</sub>* (*OpenAPI 3.1<sub>G</sub>*) e ne registra le risposte. I risultati possono

essere utilizzati uno ad uno o registrati su un apposito archivio. Tra i requisiti opzionali è descritta una proposta per gestire in modo completo questa fase.

3. **Programma di valutazione della correttezza / verosimiglianza delle risposte ricevute:** confronto tra risposte attese e risposte ottenute attraverso metodi tradizionali (es. *BM25<sub>G</sub>*), tecniche semantiche e/o confronto mediato da un LLM.
4. **Procedura di presentazione dei risultati dell'esecuzione del test:** visualizzazione sintetica e dettagliata dei risultati (statistiche generali, errori, confronti testuali).
5. **I punti precedenti devono essere integrati in un unico sistema che permetta di utilizzarli come parte di un insieme e non forniti come utility separate:** dovrà essere costruito un sistema di aggregazione dei risultati con un indice sintetico della bontà generale di quanto ottenuto, una evidenziazione dei casi migliori e peggiori e la possibilità di verificare tutte le risposte ottenute. Di conseguenza, tutte le funzionalità devono essere contenute in un'unica piattaforma coerente e facilmente utilizzabile.

### 2.2.2. Requisiti opzionali suggeriti:

- **Archiviazione dei risultati di test:** dopo una esecuzione del test è interessante poter archiviare il risultato per poi confrontarlo con altre esecuzioni. Questo richiede una gestione delle esecuzioni con la possibilità di salvare, ricercare, cancellare il documento prodotto da una esecuzione.
- **Confronto automatico tra run di test diverse:** il primo confronto sarà dato dall'indice sintetico e dalle altre valutazioni, disponendo dei dati di una esecuzione precedente può essere sviluppato un programma di confronto per evidenziare quali parti sono migliorate e quali sono peggiorate. In questo modo si ha una valutazione dei risultati che permette di capire meglio come ogni scelta nel programma che espone le API ha influenzato il risultato finale.
- **Archiviazione di set di domande/risposte:** per come sono stati proposti i requisiti obbligatori viene richiesto un unico *set di domande/risposte<sub>G</sub>* che viene eseguito nella sua interezza. Un'interessante estensione è avere un archivio che possa gestire più set di domande/risposte per verifiche di aspetti diversi del sistema o di più di un sistema.
- **Gestione della configurazione della chiamate alle API esterne:** i programmi espongono API secondo le loro regole, per poter testare vari sistemi è opportuno poter configurare il punto di chiamata delle API in modo da potersi adeguare a prodotti diversi.

## 2.3. Caratteristiche utente

L'applicazione Artificial QI è destinata a un'utenza specializzata nell'ambito dello sviluppo software e dell'integrazione di modelli di Intelligenza Artificiale, in particolare Large Language Models (*LLM<sub>G</sub>*). Sebbene non sia progettata per un uso commerciale diretto, il sistema ha finalità sperimentali e valutative, ed è pensato per assistere gli sviluppatori nella fase di validazione delle risposte generate da modelli linguistici.

### 2.3.1. Utenti del nostro prodotto

Gli utenti possono essere classificati nelle seguenti categorie:

1. **Utente sviluppatore/tecnico esperto:** È l'utente primario del sistema. Si tratta di un professionista o studente con competenze in:
  - sviluppo software e architetture web;
  - tecnologie *AI<sub>G</sub>/ML<sub>G</sub>* e utilizzo di LLM tramite API (es. *OpenAI<sub>G</sub>, Mistral<sub>G</sub>*);

- linguaggi di scripting e automazione di test;
- progettazione e *analisi dei requisiti*.

Questa tipologia di utente utilizzerà la piattaforma per testare le risposte di uno o più LLM in scenari specifici, confrontare le uscite dei modelli con le risposte attese, e valutare l'effetto delle scelte progettuali.

2. **Utente aziendale/valutatore di sistema:** Un utente secondario è rappresentato da figure aziendali o accademiche interessate alla valutazione comparativa di modelli linguistici per scopi di ricerca o adozione industriale. Pur non interagendo direttamente con il sistema a livello di configurazione tecnica, questo utente potrà:

- esaminare i risultati dei test effettuati;
- visualizzare indici di performance aggregati;
- comprendere il comportamento dei modelli sotto test, anche senza conoscenze approfondite di programmazione.

L'interfaccia dell'applicazione dovrà quindi garantire un buon equilibrio tra completezza funzionale e usabilità, permettendo a entrambi i profili di fruire delle funzionalità in modo coerente e fluido.

3. **Utente LLM (entità automatica):** Infine, un "utente" non umano ma funzionale al sistema è rappresentato dal modello LLM esterno con cui l'applicativo interagisce. Il modello:

- riceve le domande tramite chiamate API;
- fornisce risposte che verranno confrontate con quelle attese;
- viene monitorato per verificarne coerenza, affidabilità e variazioni di comportamento.

Sebbene non sia un utente nel senso tradizionale, l'LLM è un attore cruciale all'interno del sistema e le sue prestazioni costituiscono l'oggetto dell'analisi svolta dalla piattaforma.

## 2.4. Tecnologie ed Analisi della struttura di progetto

La realizzazione della piattaforma **Artificial QI** richiede l'integrazione di componenti software orientati alla valutazione delle risposte fornite da modelli linguistici esterni ( $LLM_G$ ) attraverso un'interfaccia utente accessibile, strumenti di automazione e componenti server per la gestione delle esecuzioni.

Il progetto sarà sviluppato secondo una struttura modulare, con particolare attenzione alla manutenibilità e alla possibilità di estendere le funzionalità in fasi successive ( $PoC_G$ ,  $MVP_G$ , prodotto finale). Si prevede l'utilizzo di tecnologie moderne per garantire portabilità, scalabilità e coerenza tra ambienti di sviluppo e produzione.

## 3. Casi d'uso

### 3.1. Obiettivi

Un *caso d'uso*<sub>G</sub> rappresenta formalmente una funzionalità del sistema, illustrando l'interazione tra uno o più attori esterni e il sistema stesso per il raggiungimento di uno scopo definito. Citando la presentazione P2 della sezione Riferimenti informativi, un caso d'uso è definito come segue:

«Un caso d'uso è un insieme di scenari (sequenze di azioni) che hanno in comune uno scopo finale (obiettivo) per un utente (attore)».

Questa sezione ha lo scopo di descrivere in maniera dettagliata i casi d'uso identificati a seguito dell'analisi del capitolato fornito da Zucchetti S.p.A., dei colloqui col proponente e delle discussioni interne al gruppo di progetto.

I casi d'uso costituiscono la base per l'individuazione dei requisiti funzionali e guidano le successive fasi di progettazione, implementazione e validazione del sistema.

### 3.2. Introduzione ai Casi d'Uso

Ogni caso d'uso sarà descritto secondo la seguente struttura:

- **Attori coinvolti:** entità esterne che interagiscono con il sistema;
- **Descrizione:** breve spiegazione del contesto e degli obiettivi del caso d'uso;
- **Pre-condizioni:** stato del sistema necessario per avviare l'interazione;
- **Post-condizioni:** stato del sistema al termine dell'interazione;
- **Scenario principale:** sequenza tipica di azioni svolte dall'attore e dal sistema;
- **Inclusioni/Estensioni/Generalizzazioni** (se applicabili):
  - **Inclusione** = Funzionalità comune fra più use case;
  - **Estensione** = Aumento delle funzionalità di un use case;
  - **Generalizzazione** = Aggiungere o modificare caratteristiche base.

L'identificatore dei casi d'uso seguirà il prefisso UC (Use Case), seguito da un numero naturale, eventualmente seguito da numerazione gerarchica (es. UC1.2, UC1.2.1).

### 3.3. Attori

Nel capitolato **Artificial QI** si identificano tre attori principali:

#### 1. Utente sviluppatore (principale):

È l'attore primario del sistema. Rappresenta la persona (sviluppatore/tecnico o aziendale/accademico) che utilizza l'applicazione per caricare domande e risposte attese, configurare le API dei modelli LLM da testare, avviare le sessioni di valutazione e analizzare i risultati.

*Non è previsto un sistema di autenticazione, poiché l'applicativo ha finalità puramente sperimentali e si prevede un utilizzo locale o comunque controllato.*

#### 2. LLM esterno (attore secondario):

Il modello linguistico (es. *ChatGPT*<sub>G</sub>, *Mistral*<sub>G</sub>, *LLaMA*<sub>G</sub>) rappresenta un attore passivo ma centrale nel processo. Riceve richieste di input in linguaggio naturale e produce una risposta da confrontare con quella attesa.

#### 3. Sistema di valutazione automatica (attore funzionale):

Si tratta di un modulo interno che svolge una funzione critica di valutazione ma che, per coerenza modellistica, viene trattato come attore funzionale: riceve le due risposte (attesa e ottenuta) ed esegue il confronto semantico o statistico per determinare il grado di corrispondenza.

### 3.4. Elenco dei Casi d'Uso

#### 3.4.1. UC: Gestione domande e risposte attese

##### 3.4.1.1. UC1: Visualizzazione lista domande

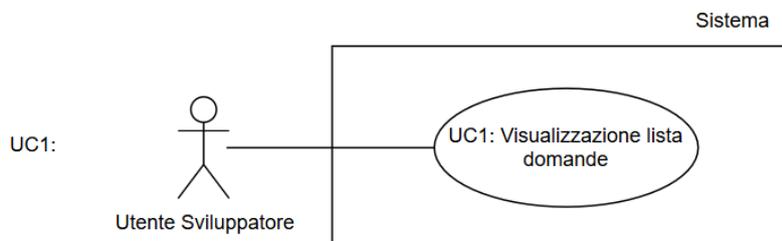


Figura 1: UC1 - Visualizzazione lista domande

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente visualizza la lista di tutte le domande presenti nel sistema.

**Pre-condizioni:** L'utente deve entrare nella sezione di «Gestione domande»

**Post-condizioni:** Viene visualizzata una lista con gli elementi domanda presenti o una lista vuota

**Scenario principale:**

1. L'utente entra nella sezione «Gestione domande»
2. L'utente visualizza la lista di domande presenti

##### 3.4.1.1.1. UC1.1: Visualizzazione singola domanda

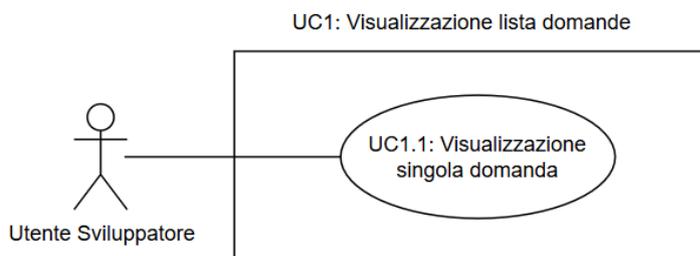


Figura 2: UC1.1 - Visualizzazione singola domanda

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente vuole visualizzare una singola domanda già esistente con la risposta attesa associata.

**Pre-condizioni:** Almeno una domanda presente nel sistema.

**Post-condizioni:** L'utente visualizza la domanda con la risposta attesa associata e la categoria di appartenenza.

**Scenario principale:**

1. L'utente seleziona la domanda.
2. L'utente visualizza la domanda con la risposta attesa associata e la categoria di appartenenza.

#### 3.4.1.1.1. UC1.1.1: Modifica domanda

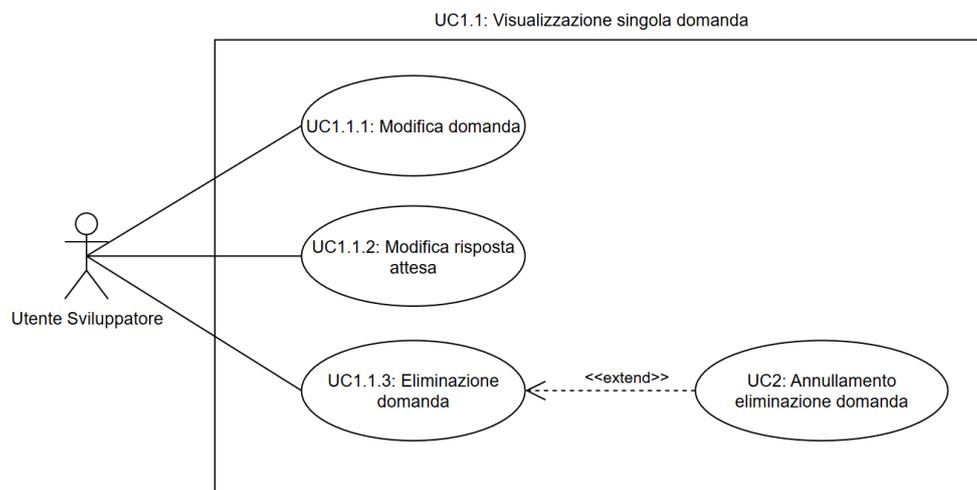


Figura 3: UC1.1.1 - Modifica domanda

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente modifica il testo della domanda selezionata.

**Pre-condizioni:** Almeno una domanda presente nel sistema.

**Post-condizioni:** Il testo della domanda viene aggiornato.

**Scenario principale:**

1. L'utente seleziona la domanda.
2. Modifica il testo.
3. Salva le modifiche.

**3.4.1.1.1.2. UC1.1.2: Modifica risposta attesa**

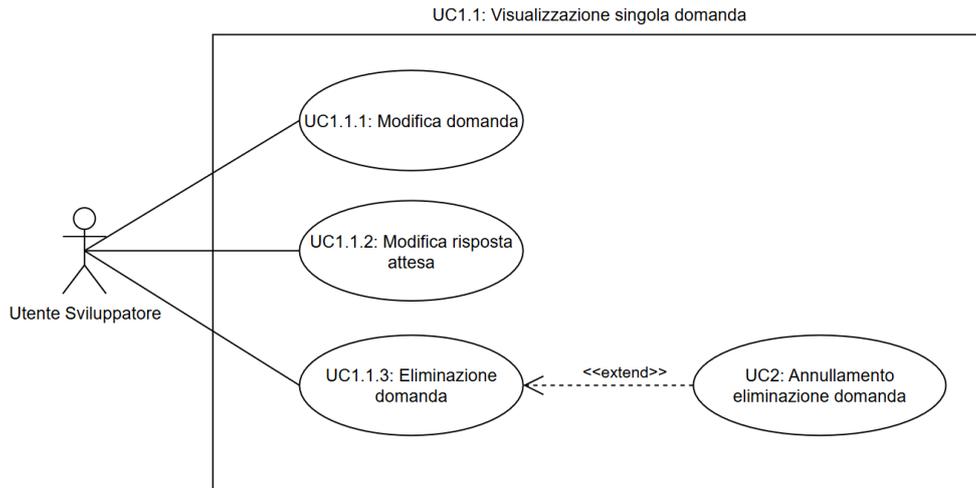


Figura 4: UC1.1.2 - Modifica risposta attesa

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente modifica il testo di una risposta attesa associata alla domanda selezionata.

**Pre-condizioni:** Almeno una domanda presente nel sistema.

**Post-condizioni:** La risposta viene aggiornata.

**Scenario principale:**

1. L'utente seleziona la domanda.
2. Modifica la risposta attesa.
3. Salva le modifiche.

**3.4.1.1.1.3. UC1.1.3: Eliminazione domanda**

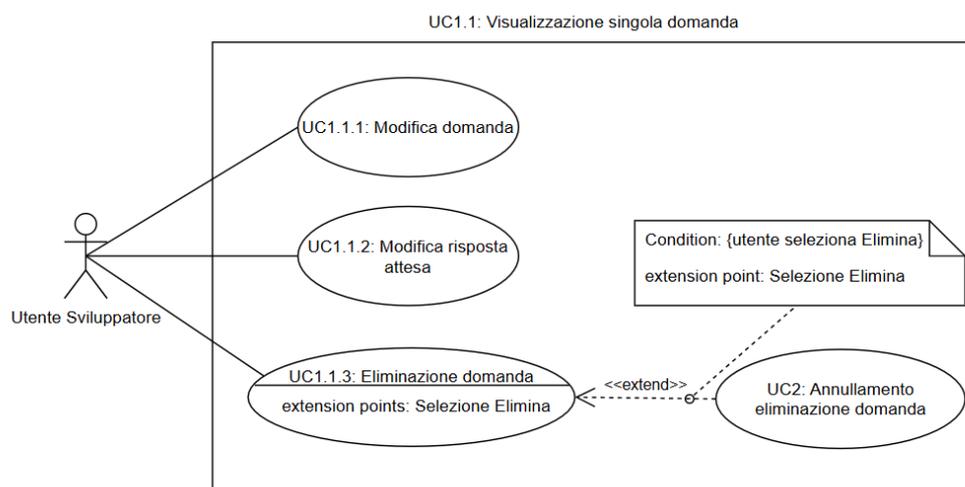


Figura 5: UC1.1.3 - Eliminazione domanda

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente elimina la domanda selezionata.

**Pre-condizioni:** Almeno una domanda nel sistema.

**Post-condizioni:** La domanda selezionata viene eliminata.

**Scenario principale:**

1. L'utente seleziona la domanda da eliminare.
2. Conferma l'azione.
3. Il sistema rimuove i dati.

### 3.4.1.2. UC2: Annullamento eliminazione domanda

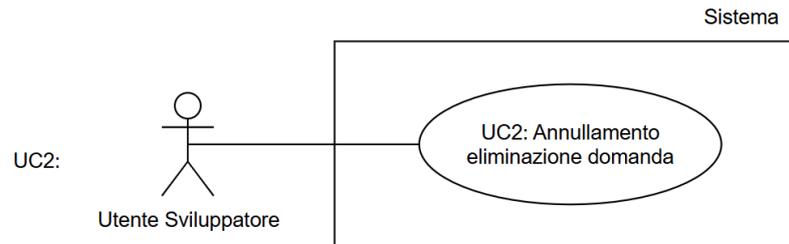


Figura 6: UC2 - Annullamento eliminazione domanda

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente, dopo aver avviato la procedura di eliminazione di una domanda, decide di annullarla prima della conferma definitiva.

**Pre-condizioni:** L'utente ha selezionato l'eliminazione di una domanda ed è nella fase di conferma dell'eliminazione.

**Post-condizioni:** Nessuna modifica è applicata al sistema; il set rimane disponibile.

**Scenario principale:**

1. L'utente seleziona la domanda
2. L'utente seleziona il pulsante di eliminazione della domanda
3. Il sistema mostra una finestra di conferma.
4. L'utente sceglie l'opzione «Annulla» o chiude la finestra.
5. L'operazione viene annullata e la domanda non viene eliminata.

### 3.4.1.3. UC3: Filtro delle domande per Categoria di appartenenza

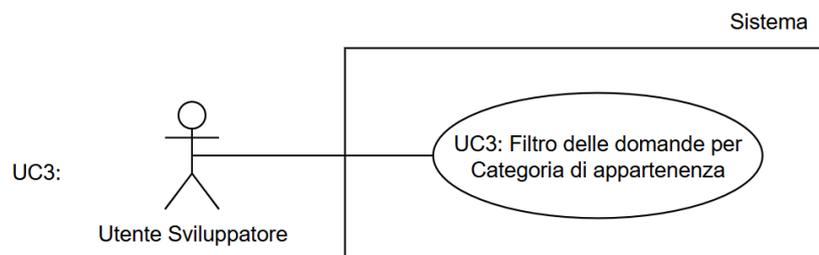


Figura 7: UC3 - Filtro delle domande per Categoria di appartenenza

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente filtra la lista delle domande per visualizzare solo quelle appartenenti ad una categoria specifica.

**Pre-condizioni:** È presente almeno una domanda nella lista, con il campo «Categoria» non vuoto.

**Post-condizioni:** Il sistema mostra solo le domande con la categoria selezionata

**Scenario principale:**

1. L'utente apre il menu dei filtri.
2. Seleziona il nome della categoria desiderata.
3. Visualizza i risultati filtrati.

#### 3.4.1.4. UC4: Aggiunta manuale domanda

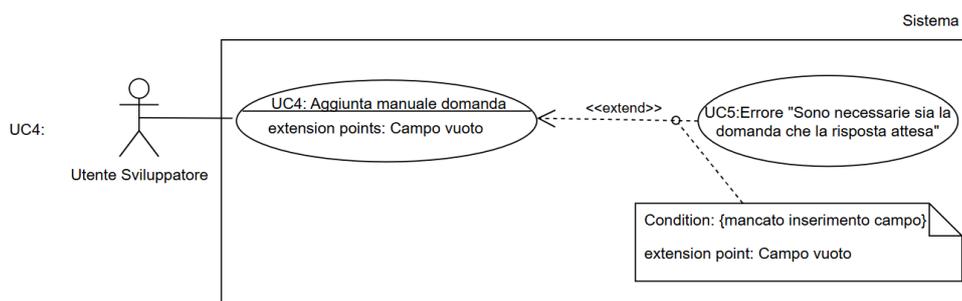


Figura 8: UC4 - Aggiunta manuale domanda

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente inserisce manualmente una domanda, la sua risposta attesa e la categoria (opzionale).

**Pre-condizioni:** L'utente ha accesso alla sezione gestione domande della webapp

**Post-condizioni:** La nuova coppia domanda/risposta è salvata.

**Scenario principale:**

1. L'utente apre il modulo di inserimento.
2. Scrive la domanda e la risposta attesa.
3. Conferma l'inserimento.

#### 3.4.1.5. UC5: Errore «Sono necessarie sia la domanda che la risposta attesa»

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente non inserisce manualmente una domanda o la sua risposta attesa (o entrambe) e ne conferma comunque l'inserimento.

**Pre-condizioni:** L'utente ha accesso alla sezione gestione domande della webapp

**Post-condizioni:** L'utente riceve un messaggio di errore che specifica che nell'inserimento di una domanda sono necessarie sia la domanda che la risposta attesa.

**Scenario principale:**

1. L'utente apre il modulo di inserimento.
2. Conferma l'inserimento senza compilare uno dei campi oppure entrambi i campi «Domanda» e «Risposta attesa».
3. L'utente riceve un messaggio di errore che specifica il motivo del mancato inserimento.

### 3.4.1.6. UC6: Aggiunta da file strutturato

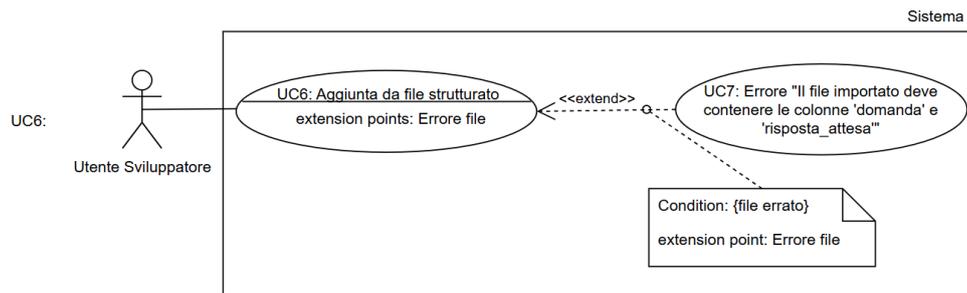


Figura 9: UC6 - Aggiunta da file strutturato

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente importa un file strutturato contenente le domande e le risposte attese.

**Pre-condizioni:** Il file è in formato valido (es. *JSON<sub>G</sub>*, *CSV<sub>G</sub>*), e rispetta le indicazioni sintattiche fornite.

**Post-condizioni:** Le domande sono salvate nel sistema.

**Scenario principale:**

1. L'utente seleziona il file da caricare.
2. Il sistema valida il formato.
3. I dati vengono salvati nel database.

### 3.4.1.7. UC7: Errore «Il file importato deve contenere le colonne “domanda” e “risposta\_attesa”»

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente importa domande/risposte da un file strutturato.

**Pre-condizioni:** L'utente ha accesso alla sezione gestione domande della webapp e inserisce tramite selezione del file un documento json/csv.

**Post-condizioni:** L'utente riceve un messaggio di errore che specifica che nell'inserimento di un file, la struttura del file deve essere come indicato nell'esempio fornito nella pagina di caricamento del file.

**Scenario principale:**

1. L'utente apre il modulo di inserimento.
2. Conferma l'inserimento del file.
3. L'utente riceve un messaggio di errore che specifica il motivo del mancato inserimento.

### 3.4.2. UC: Gestione dei set di domande

#### 3.4.2.1. UC8: Visualizzazione lista dei set

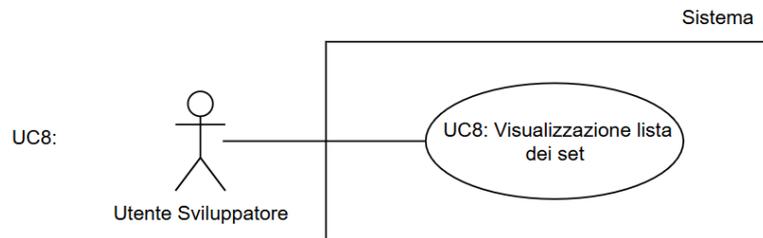


Figura 10: UC8 - Visualizzazione lista dei set

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente visualizza la lista dei set di domande salvati nel sistema per consultarli o selezionarli.

**Pre-condizioni:** Sono presenti uno o più set nel sistema.

**Post-condizioni:** I set sono visibili all'utente.

**Scenario principale:**

1. L'utente accede alla sezione "Gestione set".
2. Il sistema mostra l'elenco dei set disponibili.
3. L'utente può selezionare un set per ulteriori azioni (visualizzazione, modifica, eliminazione).

##### 3.4.2.1.1. UC8.1: Visualizzazione singolo set

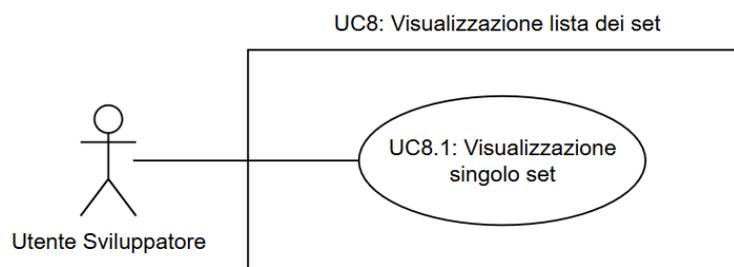


Figura 11: UC8.1 - Visualizzazione singolo set

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente vuole visualizzare un singolo set già esistente con le domande contenute al suo interno.

**Pre-condizioni:** Almeno un set presente nel sistema.

**Post-condizioni:** L'utente visualizza le domande presenti all'interno del set selezionato.

**Scenario principale:**

1. L'utente seleziona il set.
2. L'utente visualizza le domande presenti all'interno del set selezionato.

## 3.4.2.1.1.1. UC8.1.1: Modifica set

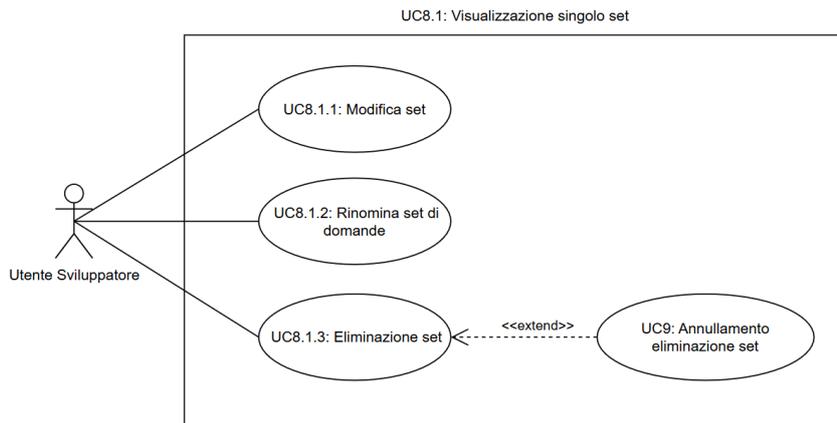


Figura 12: UC8.1.1 - Modifica set

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente modifica il contenuto di un set esistente, aggiungendo o rimuovendo domande.

**Pre-condizioni:** Il set da modificare è presente e visualizzabile.

**Post-condizioni:** Il set viene aggiornato con la nuova composizione.

**Scenario principale:**

1. L'utente seleziona un set esistente.
2. Aggiunge o rimuove domande dal set.
3. Salva le modifiche.

## 3.4.2.1.1.2. UC8.1.2: Rinomina set di domande

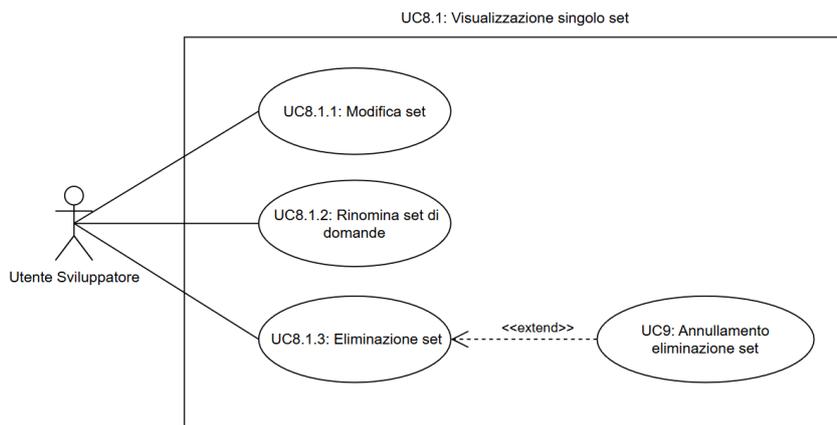


Figura 13: UC8.1.2 - Rinomina set di domande

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente modifica il nome assegnato a un set di domande.

**Pre-condizioni:** Esiste almeno un set nel sistema.

**Post-condizioni:** Il nome del set è aggiornato.

**Scenario principale:**

1. L'utente visualizza i set disponibili.
2. Seleziona il set da rinominare.
3. Inserisce il nuovo nome.
4. Conferma la modifica.

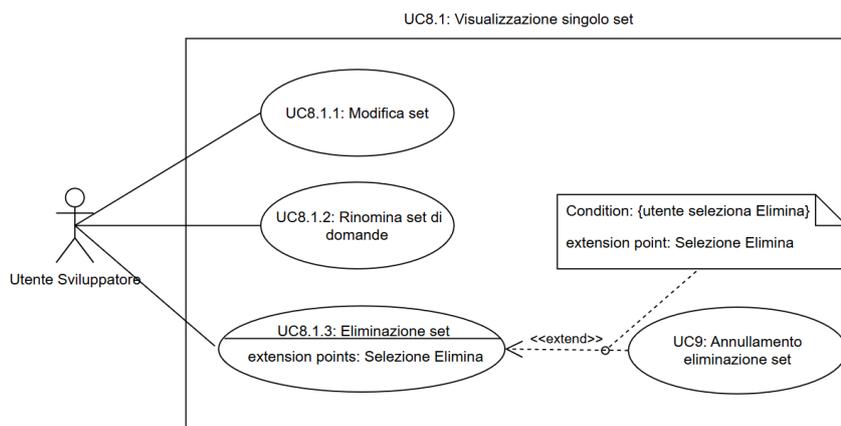
**3.4.2.1.1.3. UC8.1.3: Eliminazione set**


Figura 14: UC8.1.3 - Eliminazione set

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente elimina un set di domande dal sistema. Le domande contenute non vengono cancellate dal database.

**Pre-condizioni:** È presente almeno un set di domande.

**Post-condizioni:** Il set selezionato è rimosso.

**Scenario principale:**

1. L'utente accede alla sezione "Gestione set".
2. Seleziona un set da eliminare.
3. Conferma l'eliminazione.
4. Il sistema rimuove il set, lasciando inalterate le domande.

**3.4.2.2. UC9: Annullamento eliminazione set**

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente, dopo aver avviato la procedura di eliminazione di un set, decide di annullarla prima della conferma definitiva.

**Pre-condizioni:** L'utente ha selezionato un set ed è nella fase di conferma dell'eliminazione.

**Post-condizioni:** Nessuna modifica è applicata al sistema; il set rimane disponibile.

**Scenario principale:**

1. L'utente seleziona un set da eliminare.
2. Il sistema mostra una finestra di conferma.
3. L'utente sceglie l'opzione «Annulla» o chiude la finestra.
4. L'operazione viene annullata e il set non viene eliminato.

### 3.4.2.3. UC10: Filtro dei set per Categoria di appartenenza

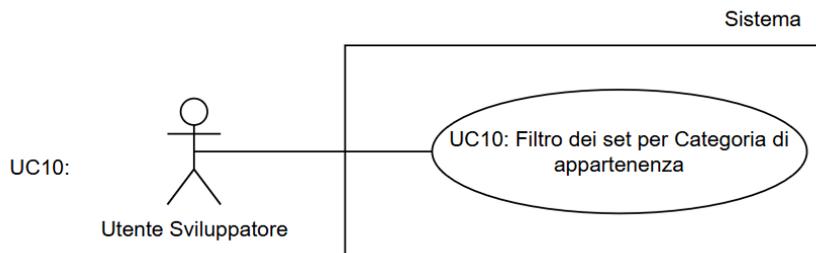


Figura 15: UC10 - Filtro dei set per Categoria di appartenenza

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente filtra la lista dei set per visualizzare solo quelli appartenenti ad una categoria specifica.

**Pre-condizioni:** È presente almeno un set nella lista.

**Post-condizioni:** Il sistema mostra solo i set con la categoria selezionata

**Scenario principale:**

1. L'utente apre il menu dei filtri.
2. Seleziona il nome della categoria desiderata.
3. Visualizza i risultati filtrati.

### 3.4.2.4. UC11: Aggiunta manuale set

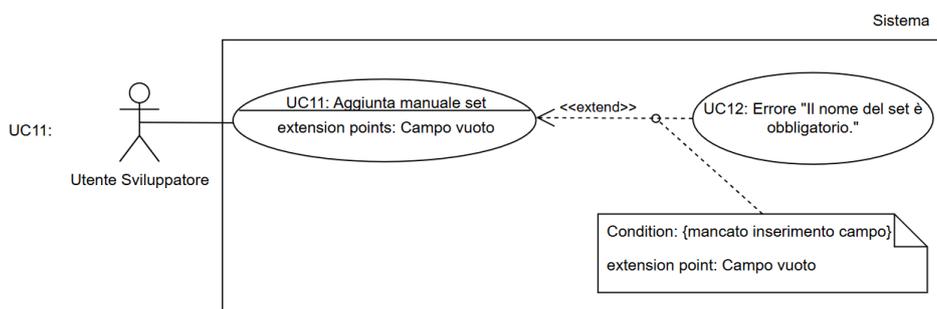


Figura 16: UC11 - Aggiunta manuale set

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente crea manualmente un set, aggiungendo le domande disponibili nel database al set.

**Pre-condizioni:** L'utente ha accesso alla sezione gestione set della webapp

**Post-condizioni:** Il nuovo set viene salvato e visualizzato correttamente.

**Scenario principale:**

1. L'utente apre il modulo di creazione set.

2. Aggiunge le domande desiderate al set.
3. Assegna un nome al set
4. Conferma la creazione di un nuovo set.

#### 3.4.2.5. UC12: Errore «Il nome del set è obbligatorio.»

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente non inserisce manualmente il nome del set e ne conferma comunque l'inserimento.

**Pre-condizioni:** L'utente ha accesso alla sezione gestione set della webapp

**Post-condizioni:** L'utente riceve un messaggio di errore che specifica che nell'inserimento di un nuovo set è necessario specificare il nome (NB: un set può essere vuoto, ma non può essere senza nome).

**Scenario principale:**

1. L'utente apre il modulo di creazione del set.
2. Conferma l'inserimento senza assegnare un nome al set.
3. L'utente riceve un messaggio di errore che specifica il motivo del mancato inserimento.

#### 3.4.2.6. UC13: Aggiunta set da file strutturato

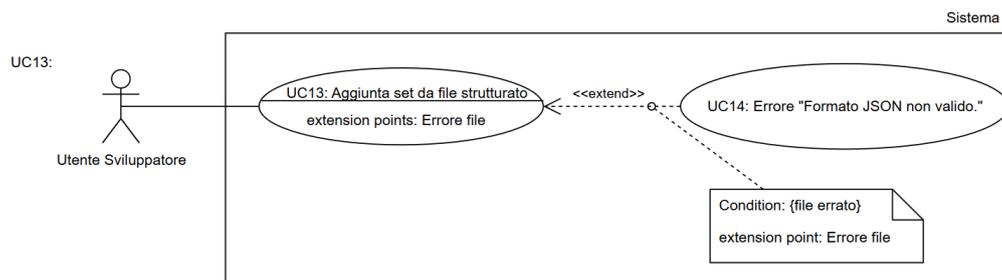


Figura 17: UC13 - Aggiunta set da file strutturato

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente importa un file strutturato contenente un set di domande già impostato.

**Pre-condizioni:** Il file è in formato valido (es. *JSON<sub>G</sub>*, *CSV<sub>G</sub>*), e rispetta le indicazioni sintattiche fornite.

**Post-condizioni:** Il set viene salvato nel sistema.

**Scenario principale:**

1. L'utente seleziona il file da caricare.
2. Il sistema valida il formato.
3. I dati vengono salvati nel database.

#### 3.4.2.7. UC14: Errore «Formato JSON non valido.»

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente importa un set da un file strutturato.

**Pre-condizioni:** L'utente ha accesso alla sezione gestione set della webapp e inserisce tramite selezione del file un documento json/csv.

**Post-condizioni:** L'utente riceve un messaggio di errore che specifica che nell'inserimento di un file, la struttura del file deve essere come indicato nell'esempio fornito nella pagina di caricamento del file.

**Scenario principale:**

1. L'utente apre il modulo di inserimento.
2. Conferma l'inserimento del file.
3. L'utente riceve un messaggio di errore che specifica il motivo del mancato inserimento.

### 3.4.3. UC: Configurazione API LLM

#### 3.4.3.1. UC15: Creazione di Preset per la Configurazione API LLM

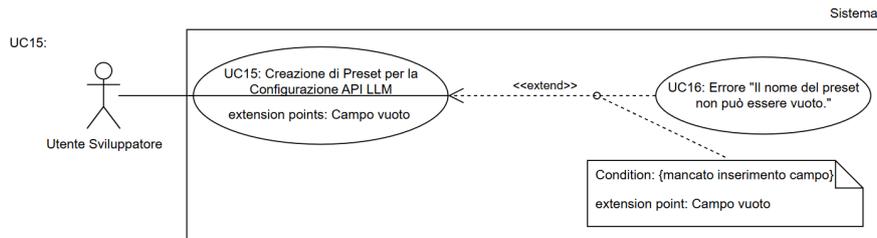


Figura 18: UC15 - Creazione di Preset per la Configurazione API LLM

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente accede alla sezione di configurazione dell'API e crea un nuovo preset.

**Pre-condizioni:** L'utente conosce i dettagli dell'API del modello (es.  $URL_G$ ,  $token_G$ ), il modello deve essere funzionante, il database deve essere impostato correttamente.

**Post-condizioni:** Il sistema crea un preset utilizzabile per effettuare test di valutazione tramite LLM.

**Scenario principale:**

1. L'utente accede alla sezione di configurazione API.
2. L'utente seleziona l'opzione di creazione nuovo preset.
3. Compila le richieste.
4. Salva la configurazione del preset.

#### 3.4.3.2. UC15.1: Test di connessione API

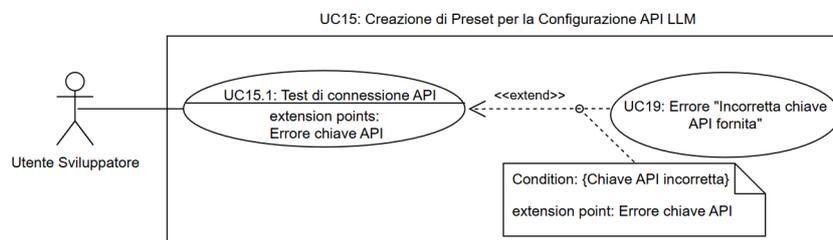


Figura 19: UC15.1 - Test di connessione API

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente al termine della compilazione dei campi richiesti per la creazione di un preset API, seleziona l'opzione di Test connessione per verificare che la connessione sia valida.

**Pre-condizioni:** L'utente accede alla compilazione del preset.

**Post-condizioni:** Il sistema restituisce una valutazione della connessione API.

**Scenario principale:**

1. L'utente accede alla sezione di configurazione API.
2. L'utente seleziona l'opzione di creazione nuovo preset.
3. Compila le richieste.
4. Esegue un test della connessione.
5. Ottiene una valutazione positiva o negativa del test.

### 3.4.3.3. UC16: Errore «Il nome del preset non può essere vuoto.»

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente compila un preset di configurazione API senza fornire un nome.

**Pre-condizioni:** L'utente ha accesso alla sezione «Configurazione API» della webapp e non inserisce un nome al preset.

**Post-condizioni:** L'utente riceve un messaggio di errore che specifica il motivo per cui il preset non è stato salvato.

**Scenario principale:**

1. L'utente apre il modulo di compilazione preset
2. Conferma l'inserimento del preset senza fornire un nome.
3. L'utente riceve un messaggio di errore che specifica il motivo del mancato inserimento.

### 3.4.3.4. UC17: Visualizzazione lista dei preset configurati

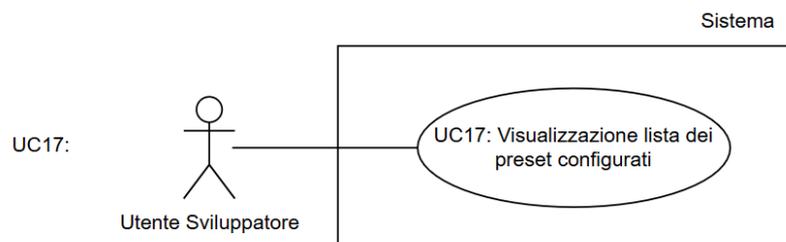


Figura 20: UC17 - Visualizzazione lista dei preset configurati

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente visualizza la lista dei preset di configurazione API per LLM salvati.

**Pre-condizioni:** Sono presenti uno o più preset nel sistema.

**Post-condizioni:** Nessuna modifica allo stato del sistema.

**Scenario principale:**

1. L'utente accede alla sezione «Configurazione API».
2. Il sistema mostra l'elenco dei preset disponibili.

### 3.4.3.4.1. UC17.1: Modifica preset

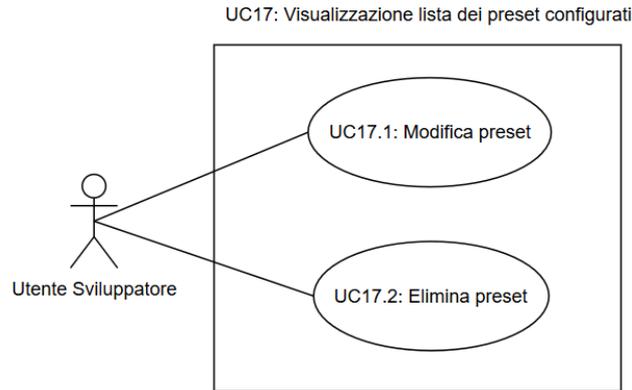


Figura 21: UC17.1 - Modifica preset

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente modifica il contenuto di un preset esistente, modificando uno o più campi (Nome preset, chiave API, temperatura...).

**Pre-condizioni:** Il preset da modificare è presente e visualizzabile.

**Post-condizioni:** Il preset viene aggiornato con la nuova composizione.

**Scenario principale:**

1. L'utente seleziona un preset esistente.
2. Modifica uno o più campi del preset.
3. Salva le modifiche.

### 3.4.3.4.2. UC17.2: Elimina preset

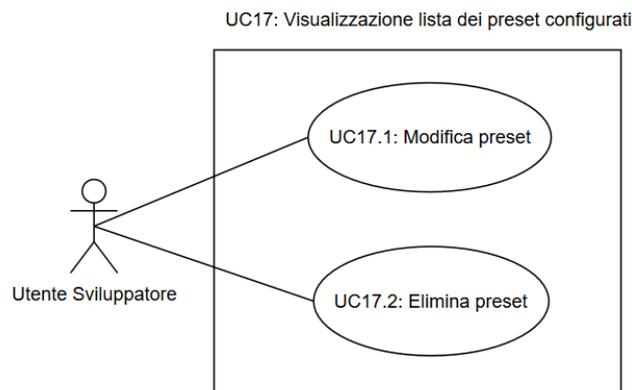


Figura 22: UC17.2 - Elimina preset

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente elimina un preset dal sistema.

**Pre-condizioni:** È presente almeno un preset nel sistema.

**Post-condizioni:** Il preset selezionato è rimosso.

**Scenario principale:**

1. L'utente accede alla sezione "Configurazione API".
2. Elimina il preset.
3. Il sistema rimuove il preset.

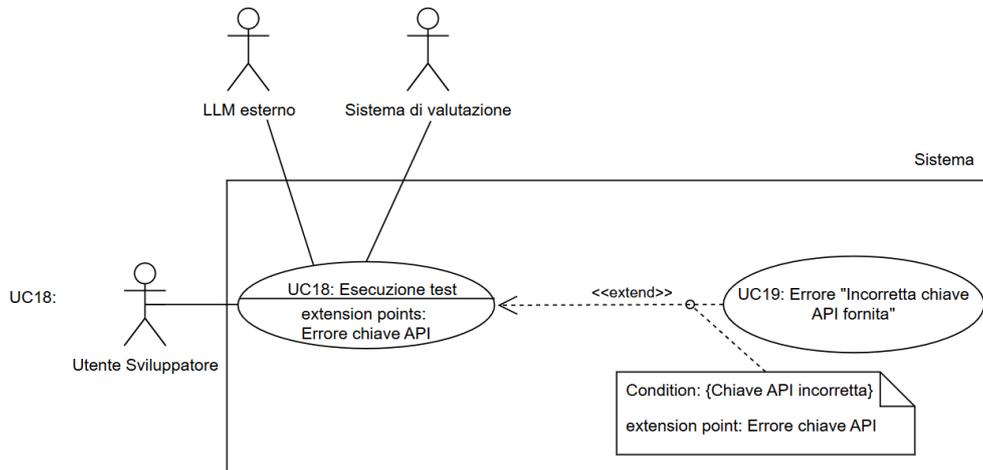
**3.4.4. UC: Avvio Test**
**3.4.4.1. UC18: Esecuzione test**


Figura 23: UC18 - Esecuzione test

**Attore principale:** Utente sviluppatore

**Attori secondari:** LLM esterno, Sistema di valutazione

**Descrizione:** L'utente seleziona un preset ed il set di domande da valutare; il sistema invia le domande all'LLM configurato nel preset, raccoglie le risposte e le confronta con quelle attese attraverso il sistema di valutazione automatica.

**Pre-condizioni:** È presente almeno una domanda con risposta attesa, è presente almeno un preset di configurazione API.

**Post-condizioni:** I risultati vengono salvati e mostrati.

**Scenario principale:**

1. L'utente seleziona un set di domande da valutare.
2. Seleziona il preset da utilizzare per la valutazione.
2. Avvia il test.
3. Il sistema invia domande all'LLM.
4. Riceve e registra le risposte.
5. Avvia la valutazione automatica.
6. Salva i risultati.

### 3.4.4.1.1. UC18.1: Visualizzazione preset di configurazione API salvati

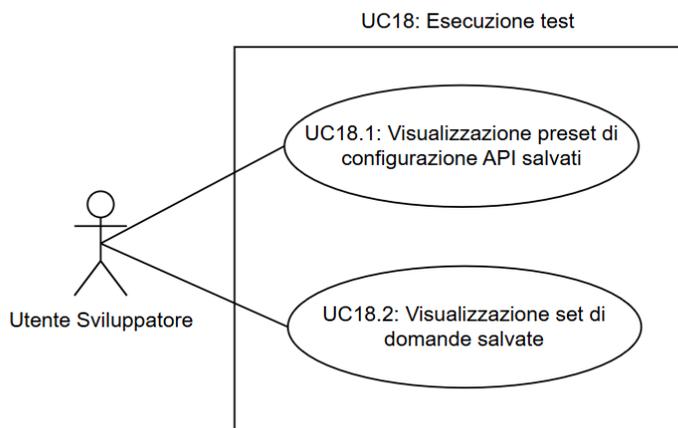


Figura 24: UC18.1 - Visualizzazione preset di configurazione API salvati

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente visualizza tutti i preset salvati all'interno della sezione di esecuzione test, in modo da poter selezionare il preset desiderato in fase di test.

**Pre-condizioni:** È disponibile almeno un preset.

**Post-condizioni:** Nessuna modifica allo stato del sistema.

**Scenario principale:**

1. L'utente accede alla sezione "Esecuzione test".
2. Il sistema mostra l'elenco dei preset disponibili.
3. L'utente può selezionare un preset per eseguire il test.

### 3.4.4.1.2. UC18.2: Visualizzazione set di domande salvate

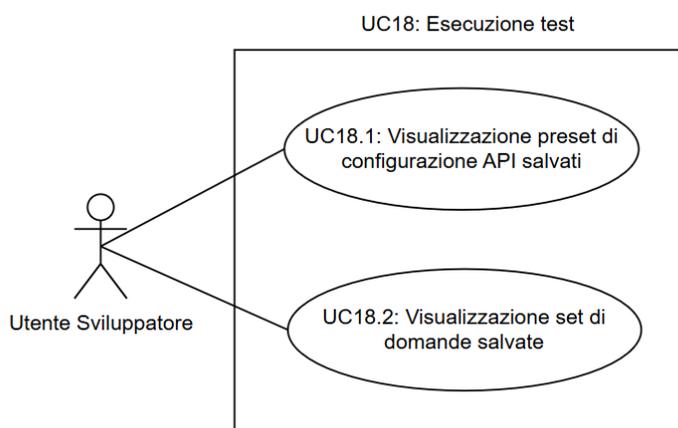


Figura 25: UC18.2 - Visualizzazione set di domande salvate

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente deve poter visualizzare tutti i set di domande salvati all'interno della sezione di esecuzione test, in modo da poter selezionare il set desiderato in fase di test.

**Pre-condizioni:** È stato caricato o creato almeno un set di domande.

**Post-condizioni:** Nessuna modifica allo stato del sistema.

**Scenario principale:**

1. L'utente accede alla sezione "Esecuzione test".
2. Il sistema mostra l'elenco dei set disponibili.
3. L'utente può selezionare un set su cui eseguire il test.

#### 3.4.4.2. UC19: Errore «Incorretta chiave API fornita»

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente avvia un test tramite un preset configurato.

**Pre-condizioni:** L'utente ha accesso alla sezione «Esecuzione test» della webapp e avvia un test.

**Post-condizioni:** L'utente riceve un messaggio di errore che specifica il motivo per cui il test non è stato eseguito.

**Scenario principale:**

1. L'utente apre il modulo di esecuzione test
2. Avvia l'esecuzione del test.
3. L'utente riceve un messaggio di errore che specifica il motivo della mancata esecuzione del test.

#### 3.4.4.3. UC20: Valutazione automatica

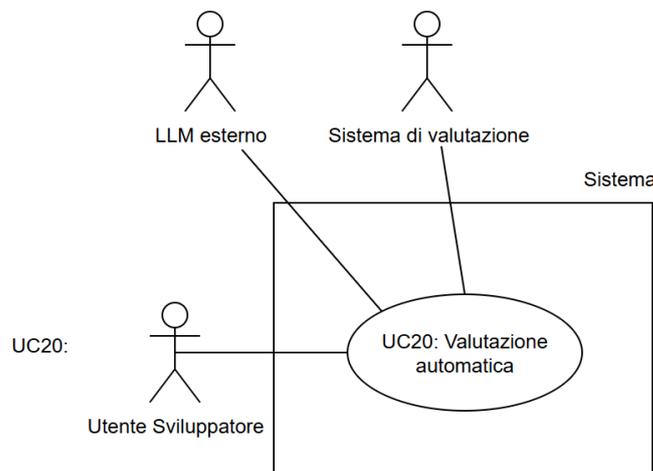


Figura 26: UC20 - Valutazione automatica

**Attore principale:** Sistema di valutazione automatica

**Attori secondari:** Utente sviluppatore, LLM esterno

**Descrizione:** Il sistema di valutazione automatica riceve una coppia di risposte (quella attesa e quella generata dall'LLM) e produce un punteggio o giudizio qualitativo in base a criteri semantici, lessicali o statistici.

**Pre-condizioni:** È disponibile una risposta attesa e una risposta generata dall'LLM.

**Post-condizioni:** Il risultato della valutazione è disponibile per la visualizzazione e l'analisi da parte dell'utente sviluppatore nella sezione «Visualizza risultati».

**Scenario principale:**

1. Il sistema riceve in input la risposta attesa e quella ottenuta.
2. Applica l'algoritmo di confronto.
3. Calcola un punteggio o una valutazione qualitativa.
4. Restituisce il risultato al sistema principale.

### 3.4.5. UC: Visualizzazione dei risultati

#### 3.4.5.1. UC21: Visualizzazione risultati

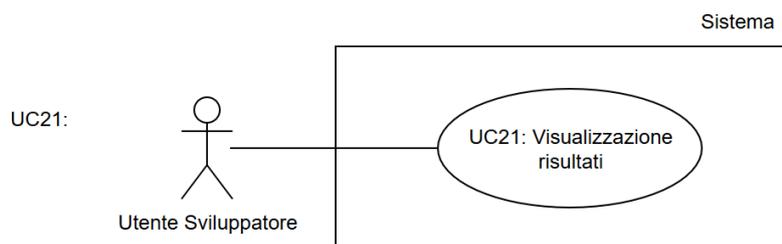


Figura 27: UC21 - Visualizzazione risultati

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente consulta i risultati ottenuti dal confronto tra risposte attese e risposte dell'LLM.

**Pre-condizioni:** È stata eseguita almeno una sessione di valutazione.

**Post-condizioni:** Nessuna modifica allo stato del sistema.

**Scenario principale:**

1. L'utente accede alla sezione di visualizzazione risultati.
2. Visualizza per ogni domanda la risposta attesa, la risposta dell'LLM e l'esito del confronto.

### 3.4.5.1.1. UC21.1: Visualizzazione risultati per singola domanda

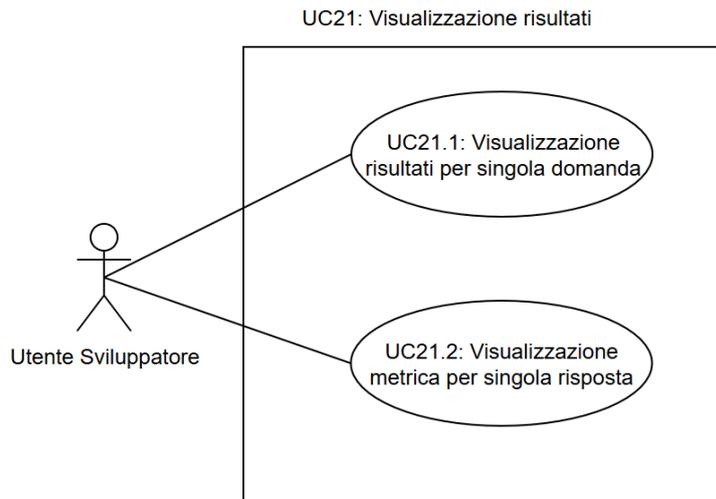


Figura 28: UC21.1 - Visualizzazione risultati per singola domanda

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente accede al dettaglio del risultato relativo a una specifica domanda, osservando risposta generata, attesa e valutazione associata.

**Pre-condizioni:** È stato eseguito almeno un test.

**Post-condizioni:** Nessuna modifica allo stato.

**Scenario principale:**

1. L'utente entra nella sezione risultati.
2. Seleziona una domanda.
3. Visualizza risposta attesa, risposta dell'LLM e valutazione.

### 3.4.5.1.2. UC21.2: Visualizzazione metrica per singola risposta

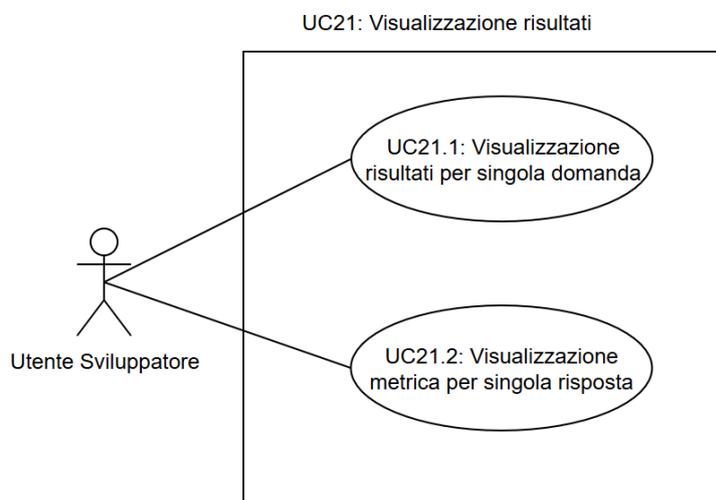


Figura 29: UC21.2 - Visualizzazione metrica per singola risposta

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente visualizza le singole metriche applicate a una risposta.

**Pre-condizioni:** La valutazione è stata eseguita.

**Post-condizioni:** Nessuna modifica allo stato.

**Scenario principale:**

1. L'utente accede al dettaglio di una domanda.
2. Visualizza la sezione "metriche".
3. Visualizza ogni metrica applicata e il relativo punteggio.

### 3.4.5.2. UC22: Salvataggio risultati del test

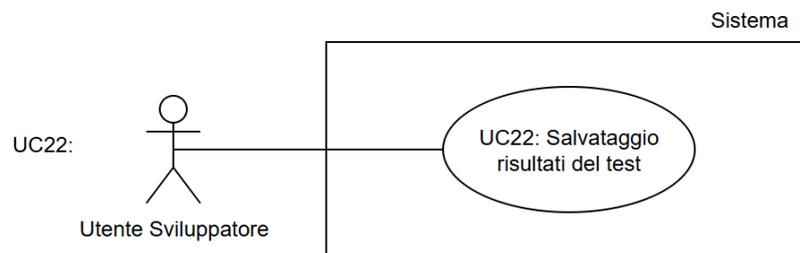


Figura 30: UC22 - Salvataggio risultati del test

**Attore principale:** Sistema

**Attori secondari:** Utente sviluppatore

**Descrizione:** Al termine di un'esecuzione, il sistema salva i risultati del test (risposte, valutazioni e metadati) per consentirne la consultazione successiva o l'esportazione.

**Pre-condizioni:** È stato eseguito un test su uno o più set.

**Post-condizioni:** I risultati vengono salvati nel sistema e associati a un identificativo univoco di esecuzione.

**Scenario principale:**

1. Il sistema riceve l'output del confronto tra risposte attese e risposte generate.
2. Per ogni domanda, associa: input, risposta LLM, valutazione e metadati.
3. Registra i dati nel database/storico.
4. Rende disponibili i risultati per UC21 (visualizzazione) e UC26 (storico).

### 3.4.5.3. UC23: Download risultati del test

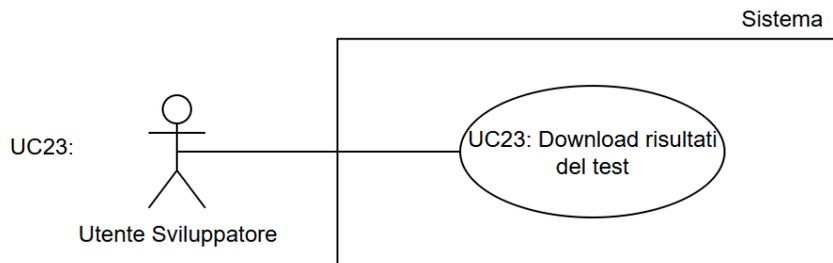


Figura 31: UC23 - Download risultati del test

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente scarica i risultati in un file strutturato contenente domande, risposte, valutazioni, metriche e metadati.

**Pre-condizioni:** È stato eseguito almeno un test.

**Post-condizioni:** Il file è scaricato localmente.

**Scenario principale:**

1. L'utente seleziona il test da esportare.
2. Sceglie nome, percorso e formato del file.
3. Avvia il salvataggio.

### 3.4.5.4. UC24: Caricamento risultati da file

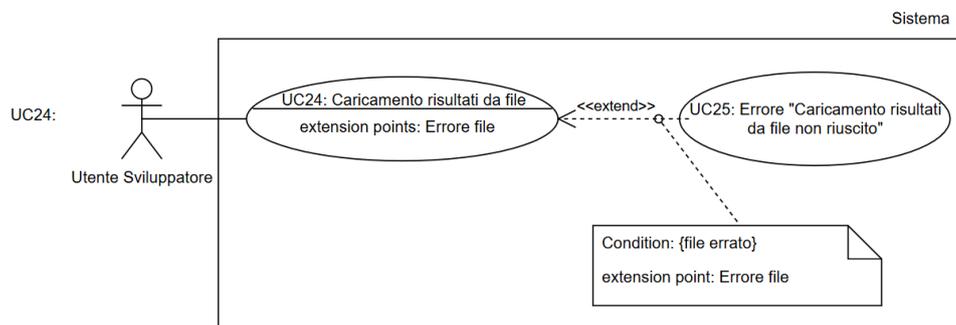


Figura 32: UC24 - Caricamento risultati da file

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente importa nel sistema i risultati di un test precedentemente salvato.

**Pre-condizioni:** Il file caricato è valido.

**Post-condizioni:** I risultati vengono aggiunti allo storico.

**Scenario principale:**

1. L'utente seleziona il file.
2. Il sistema lo analizza.
3. I dati vengono importati e associati.

### 3.4.5.5. UC25: Errore «Caricamento risultati da file non riuscito»

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente viene notificato della presenza di un errore dato da un formato errato del file

**Pre-condizioni:** L'utente carica un file contenente dei dati in formato errato.

**Post-condizioni:** L'utente riceve un messaggio di errore che indica che il file è errato.

**Scenario principale:**

1. L'utente seleziona il file.
2. Il sistema lo analizza.
3. L'utente riceve un errore a causa del formato errato dei file.

### 3.4.5.6. UC26: Visualizzazione storico dei test

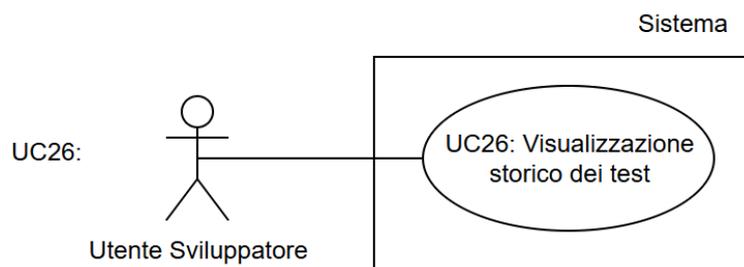


Figura 33: UC26 - Visualizzazione storico dei test

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente visualizza lo storico delle esecuzioni dei test precedentemente effettuati per poterli consultare, confrontare o rieseguire.

**Pre-condizioni:** È stato effettuato almeno un test.

**Post-condizioni:** L'elenco delle esecuzioni è visibile.

**Scenario principale:**

1. L'utente accede alla sezione "Storico test".
2. Il sistema mostra una lista cronologica o filtrabile di tutte le esecuzioni passate.
3. L'utente può selezionare un'esecuzione per visualizzarne i dettagli (UC26.1) o confrontarla (UC28).

### 3.4.5.6.1. UC26.1: Visualizzazione elemento esecuzione test

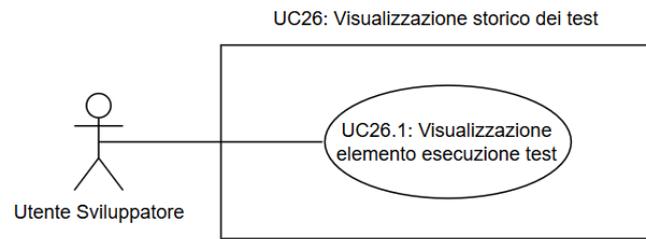


Figura 34: UC26.1 - Visualizzazione elemento esecuzione test

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente visualizza tutti i dati relativi a una specifica esecuzione, come set utilizzato, LLM testato, punteggio generale e risultati delle singole domande.

**Pre-condizioni:** Lo storico dei test è disponibile.

**Post-condizioni:** Nessuna modifica allo stato del sistema.

**Scenario principale:**

1. L'utente seleziona un test dallo storico.
2. Il sistema mostra:
  - Nome del set
  - Modello LLM utilizzato
  - Valutazione media
  - Lista di tutte le domande, risposte e valutazioni
3. L'utente può accedere al dettaglio di ogni domanda (vedi UC21.1/UC21.2).

### 3.4.5.7. UC27: Applicazione filtri storico dei test

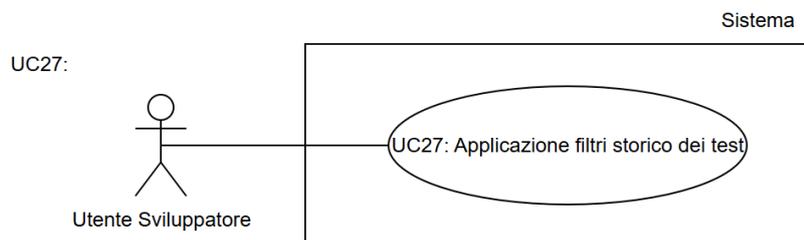


Figura 35: UC27 - Applicazione filtri storico dei test

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente filtra lo storico dei test in base a criteri come nome del set utilizzato o modello LLM impiegato.

**Pre-condizioni:** Sono presenti più esecuzioni testate nel sistema.

**Post-condizioni:** Il sistema visualizza solo i test che soddisfano i filtri applicati.

**Scenario principale:**

1. L'utente accede allo storico.
2. Seleziona uno o più criteri di filtro.
3. Il sistema aggiorna la visualizzazione.

**Generalizzazioni:**

- **UC27.1: Filtro storico dei test per nome del set**
- **UC27.2: Filtro storico dei test per LLM**
- **UC27.3: Filtro storico dei test per risultati di valutazione**

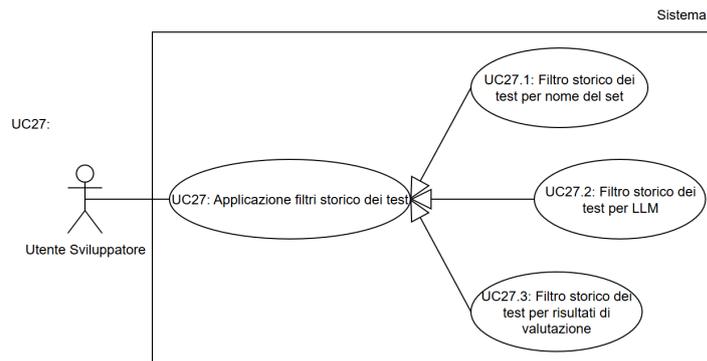
**3.4.5.7.1. UC27.1: Filtro storico dei test per nome del set**


Figura 36: UC27.1 - Filtro storico dei test per nome del set

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente filtra lo storico dei test per visualizzare solo quelli eseguiti su un determinato set di domande.

**Pre-condizioni:** Sono presenti test associati a set diversi.

**Post-condizioni:** Il sistema mostra solo i test del set selezionato.

**Scenario principale:**

1. L'utente apre il menu dei filtri.
2. Seleziona il nome del set.
3. Visualizza i risultati filtrati.

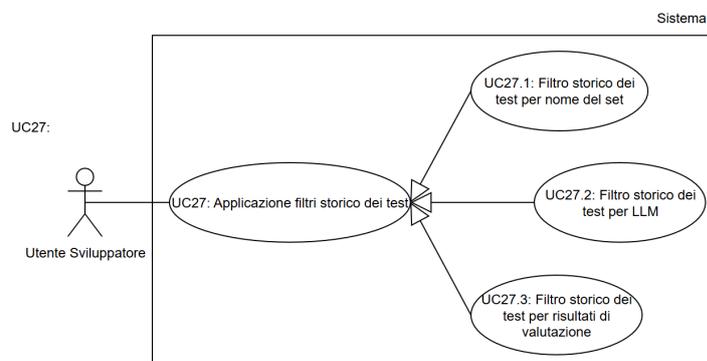
**3.4.5.7.2. UC27.2: Filtro storico dei test per LLM**


Figura 37: UC27.2 - Filtro storico dei test per LLM

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente filtra lo storico per visualizzare solo i test eseguiti con uno specifico modello LLM.

**Pre-condizioni:** Sono presenti test eseguiti con più LLM.

**Post-condizioni:** Il sistema mostra solo i test eseguiti con il modello selezionato.

**Scenario principale:**

1. L'utente apre il menu dei filtri.
2. Seleziona il nome del LLM.
3. Visualizza i test associati.

### 3.4.5.7.3. UC27.3: Filtro storico dei test per risultati di valutazione

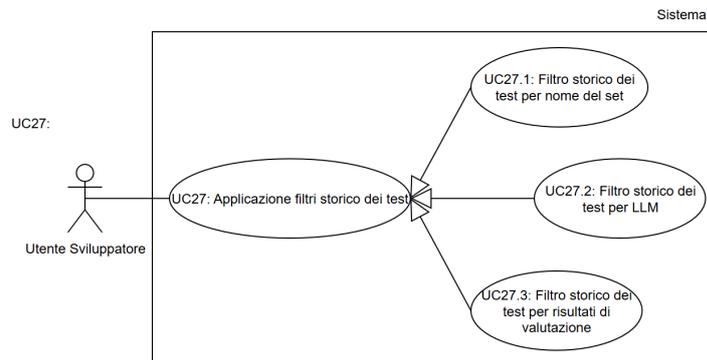


Figura 38: UC27.3 - Filtro storico dei test per risultati di valutazione

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente ordina i risultati di un test per individuare i peggiori o migliori in base alla valutazione.

**Pre-condizioni:** Esiste almeno un risultato disponibile.

**Post-condizioni:** I risultati sono visualizzati in ordine scelto.

**Scenario principale:**

1. L'utente accede ai risultati di un test.
2. Seleziona il criterio di ordinamento (es. punteggio crescente).
3. Visualizza l'elenco ordinato.

### 3.4.5.8. UC28: Confronto tra esecuzioni diverse di test

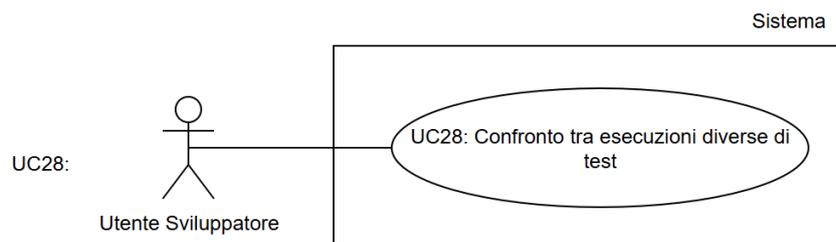


Figura 39: UC28 - Confronto tra esecuzioni diverse di test

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente seleziona due esecuzioni differenti per confrontarne i risultati, al fine di valutarne differenze e variazioni.

**Pre-condizioni:** Sono disponibili almeno due esecuzioni.

**Post-condizioni:** Il sistema genera un confronto visualizzabile.

**Scenario principale:**

1. L'utente accede allo storico.
2. Seleziona due esecuzioni da confrontare.
3. Il sistema avvia il confronto ed elabora le differenze.
4. Viene mostrato il risultato (UC29).

### 3.4.5.9. UC29: Visualizzazione risultato confronto esecuzioni

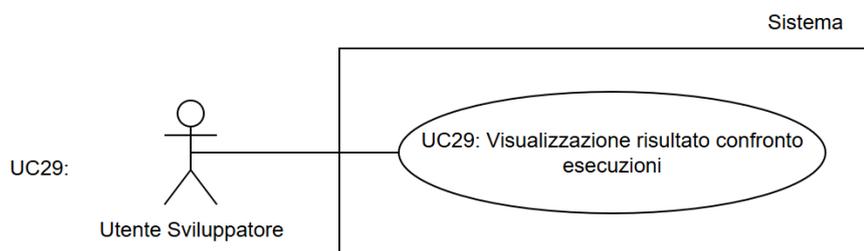


Figura 40: UC29 - Visualizzazione risultato confronto esecuzioni

**Attore principale:** Utente sviluppatore

**Descrizione:** Il sistema mostra un confronto tra due test, evidenziando variazioni di valutazione tra le domande comuni.

**Pre-condizioni:** Due test sono stati selezionati per il confronto.

**Post-condizioni:** Il confronto è visualizzato all'utente.

**Scenario principale:**

1. Il sistema confronta domande comuni tra i due test.
2. Per ogni domanda visualizza:
  - Risposte generate nei due test
  - Differenza nei punteggi
  - Metriche dettagliate
3. L'utente può accedere al dettaglio (UC29.1).

### 3.4.5.9.1. UC29.1: Visualizzazione elemento confronto esecuzione

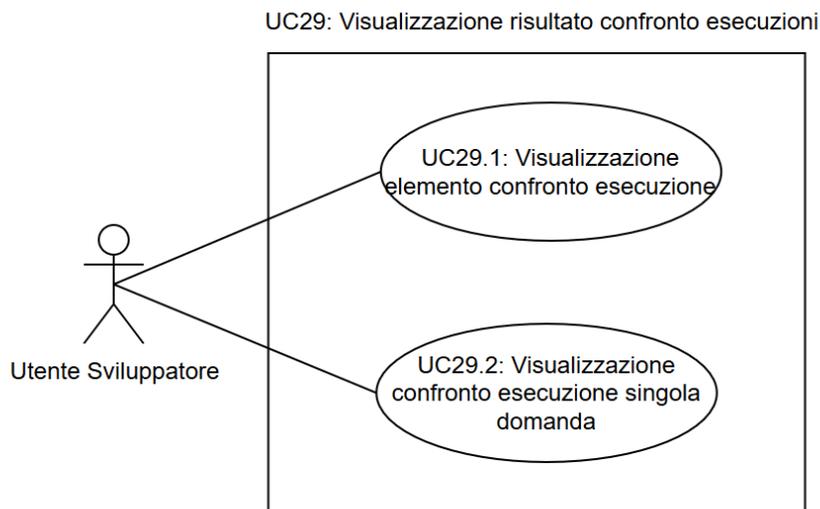


Figura 41: UC29.1 - Visualizzazione elemento confronto esecuzione

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente osserva il confronto tra due risposte LLM per una singola domanda specifica, incluso il dettaglio delle metriche.

**Pre-condizioni:** Il confronto tra test è attivo (UC28).

**Post-condizioni:** Nessuna modifica allo stato.

**Scenario principale:**

1. L'utente seleziona una domanda tra quelle comuni.
2. Il sistema mostra:
  - Risposta dell'LLM A
  - Risposta dell'LLM B
  - Valutazioni e differenze
  - Valori delle metriche

### 3.4.5.9.2. UC29.2: Visualizzazione confronto esecuzione singola domanda

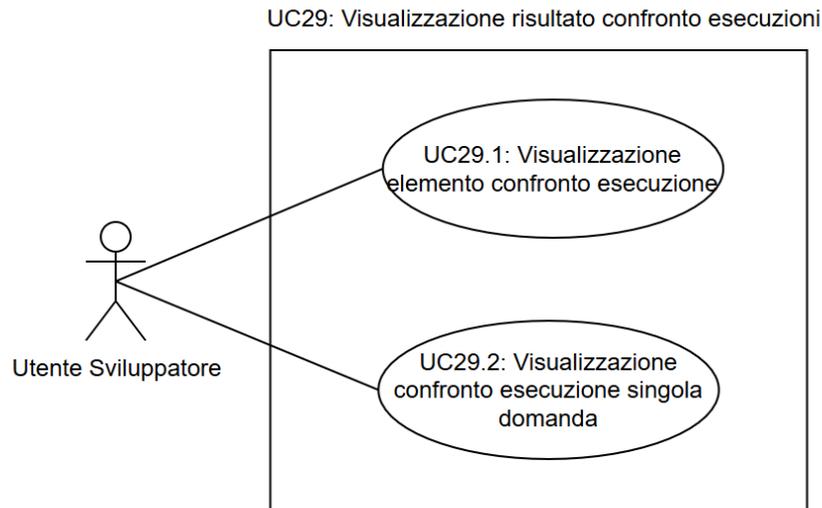


Figura 42: UC29.2 - Visualizzazione confronto esecuzione singola domanda

**Attore principale:** Utente sviluppatore

**Descrizione:** L'utente esplora nel dettaglio l'evoluzione di una singola domanda su più esecuzioni, utile per tracciare il comportamento nel tempo.

**Pre-condizioni:** La domanda è stata usata in più test.

**Post-condizioni:** Viene mostrato l'andamento storico della valutazione per la domanda.

**Scenario principale:**

1. L'utente cerca o seleziona una domanda.
2. Il sistema recupera tutti i test in cui è stata usata.
3. Visualizza in ordine cronologico:
  - Modello LLM utilizzato
  - Risposta generata
  - Valutazione e metrica
  - Eventuali variazioni di contesto

## 4. Requisiti

Ogni *requisito*<sub>C</sub> analizzato sarà identificato univocamente da una sigla del tipo **R**[**Tipo**][**Priorità**][**Codice**] nella quale:

- **R** sta per Requisito;
- [**Tipo**] può essere:
  - **F** se Funzionale;
  - **Q** se di Qualità;
  - **V** se di Vincolo.
- [**Priorità**] può essere:
  - **O** per Obbligatorio;
  - **D** per Desiderabile;
  - **P** per Opzionale.
- [**Codice**]: identifica univocamente i requisiti per ogni tipologia. È un numero intero progressivo univoco assegnato in ordine di importanza se il *requisito*<sub>C</sub> non ha padre, se invece si tratta di un sotto-requisito segue il formato [**Codice padre**].[**Numero figlio**] e trattandosi di una struttura ricorsiva non c'è limite alla profondità della gerarchia. I codici sono numerati in base alla sezione e alla classificazione (Es: RFO01 = Requisito Funzionale Obbligatorio 1, RQO01 = Requisito di Qualità Obbligatorio 1, RVO01 = Requisito di Vincolo Obbligatorio 1).

### 4.1. Requisiti di Funzionalità

Codice	Descrizione	Classificazione	Fonti
RFO01	L'utente deve poter visualizzare la lista delle domande	Obbligatorio	UC1
RFO02	L'utente deve poter visualizzare una singola domanda	Obbligatorio	UC1.1
RFO03	L'utente deve poter modificare una singola domanda	Obbligatorio	UC1.1.1
RFO04	L'utente deve poter modificare una risposta attesa	Obbligatorio	UC1.1.2
RFO05	L'utente deve poter eliminare una singola domanda	Obbligatorio	UC1.1.3
RFD06	L'utente vuole poter annullare l'operazione di eliminazione di una singola domanda	Desiderabile	UC2
RFD07	L'utente deve poter filtrare le domande in base alla categoria di appartenenza	Desiderabile	UC3
RFO08	L'utente deve essere avvisato nel caso in cui l'aggiunta manuale della domanda non sia corretta e/o completa	Obbligatorio	UC5
RFO09	L'utente deve poter aggiungere manualmente una domanda	Obbligatorio	UC4
RFO10	L'utente deve poter aggiungere domande e relative risposte attese tramite un file	Obbligatorio	UC6

RFO11	L'utente vuole poter essere notificato nel caso in cui il file di importazione delle domande non sia corretto	Obbligatorio	UC7
RFO12	L'utente deve poter visualizzare la lista dei set di domande salvati nel sistema	Obbligatorio	UC8
RFO13	L'utente deve poter visualizzare un singolo set di domande	Obbligatorio	UC8.1
RFO14	L'utente deve poter modificare un singolo set di domande	Obbligatorio	UC8.1.1
RFO15	L'utente deve poter rinominare un set di domande	Obbligatorio	UC8.1.2
RFO16	L'utente deve poter eliminare un set di domande	Obbligatorio	UC8.1.3
RFD17	L'utente vuole poter annullare l'operazione di eliminazione di un set di domande	Desiderabile	UC9
RFD18	L'utente deve poter filtrare i set di domande in base alla categoria di appartenenza	Desiderabile	UC10
RFP19	L'utente deve poter aggiornare manualmente un set di domande	Opzionale	UC11
RFO20	L'utente vuole poter essere notificato nel caso in cui non inserisca correttamente il nome del set di domande	Obbligatorio	UC12
RFO21	L'utente deve poter aggiungere un set di domande da file strutturato	Obbligatorio	UC13
RFO22	L'utente vuole poter essere notificato nel caso in cui il file contenente il set da aggiungere non sia valido	Obbligatorio	UC14
RFO23	L'utente deve poter accedere alla sezione di configurazione dell'API e creare un preset	Obbligatorio	UC15
RFO24	L'utente deve poter essere notificato nel caso in cui il nome del preset non sia stato inserito	Obbligatorio	UC16
RFP25	L'utente deve poter effettuare un test di connessione dopo aver compilato i campi necessari per la creazione di un preset API	Opzionale	UC15.1
RFP26	L'utente deve poter essere notificato nel caso in cui l'esito del test sia negativo	Opzionale	UC19

RFO27	L'utente deve poter visualizzare la lista dei preset che sono stati configurati	Obbligatorio	UC17
RFO28	L'utente deve poter modificare un preset precedentemente configurato	Obbligatorio	UC17.1
RFO29	L'utente deve poter eliminare un preset precedentemente configurato	Obbligatorio	UC17.2
RFO30	Il sistema deve poter inviare domande a un LLM esterno tramite API e riceverne le risposte.	Obbligatorio	UC18 Capitolato Zucchetti
RFO31	L'utente deve poter visualizzare tutti i preset salvati all'interno della sezione di esecuzione test	Obbligatorio	UC18.1
RFO32	L'utente deve poter visualizzare tutti i set di domande salvati all'interno della sezione di esecuzione test	Obbligatorio	UC18.2
RFO33	L'utente deve poter visualizzare il punteggio o il giudizio qualitativo assegnato dal sistema in seguito al confronto tra la risposta attesa e la risposta generata dall'LLM	Obbligatorio	UC20 Capitolato Zucchetti
RFO34	L'utente deve poter visualizzare i risultati ottenuti dal confronto tra risposte attese e risposte dell'LLM	Obbligatorio	UC21 Capitolato Zucchetti
RFO35	L'utente deve poter visualizzare i risultati ottenuti dal confronto tra risposte attese e risposte dell'LLM relativi ad una singola domanda	Obbligatorio	UC21.1
RFO36	L'utente deve poter visualizzare le metriche applicate ad una singola risposta	Obbligatorio	UC21.2
RFO37	L'utente deve poter salvare e consultare i risultati del test (risposte, valutazioni e metadati)	Obbligatorio	UC22 Capitolato Zucchetti
RFP38	L'utente deve poter scaricare i risultati di un test	Opzionale	UC23 Capitolato Zucchetti
RFP39	L'utente deve poter caricare nel sistema un file di risultati di test effettuati precedentemente	Opzionale	UC24 Capitolato Zucchetti
RFO40	L'utente vuole poter essere notificato nel caso in cui il caricamento del file contenete i risultati non vada a buon fine	Obbligatorio	UC25
RFO41	L'utente deve poter visualizzare i test effettuati precedentemente	Obbligatorio	UC26 Capitolato Zucchetti

RFP42	L'utente deve poter visualizzare tutti i dettagli di un determinato test effettuati precedentemente	Opzionale	UC26.1
RFD43	L'utente deve poter filtrare lo storico dei test in base a diversi criteri	Desiderabile	UC27, UC27.1, UC27.2, UC27.3
RFP44	L'utente deve poter confrontare due diverse esecuzioni di test	Opzionale	UC28, UC29 Capitolato Zucchetti
RFP45	L'utente deve poter confrontare due diverse risposte LLM per una singola domanda specifica	Opzionale	UC29.1, UC29.2

## 4.2. Requisiti di Qualità

Codice	Descrizione	Classificazione	Fonti
RQO01	Il sistema deve fornire un'interfaccia intuitiva per utenti non esperti in UI.	Obbligatorio	Capitolato Zucchetti
RQO02	Il sistema deve restituire feedback chiari sulle operazioni utente (salvataggi, errori, ecc.).	Obbligatorio	UC2, UC5, UC7, UC9, UC12, UC14, UC16, UC19, UC25 Capitolato Zucchetti
RQD03	Il sistema deve garantire affidabilità salvando in modo persistente tutti i dati dei test.	Desiderabile	UC22, UC23 Capitolato Zucchetti
RQP04	Il sistema deve supportare l'esportazione e importazione di risultati test in vari formati.	Opzionale	UC23, UC24 Capitolato Zucchetti
RQP05	Il sistema deve consentire di effettuare test multipli senza perdita di dati, anche in caso di errore o annullamento.	Opzionale	UC18, UC20
RQD06	Il sistema deve supportare il filtraggio e la ricerca rapida.	Desiderabile	UC3, UC10, UC21, UC21.1, UC21.2, UC27, UC27.1, UC27.2, UC27.3
RQO07	Il sistema deve adottare una struttura modulare e manutenibile per la gestione di LLM multipli.	Obbligatorio	Capitolato Zucchetti

### 4.3. Requisiti di Vincolo

Codice	Descrizione	Classificazione	Fonti
RVO01	Il sistema deve essere sviluppato come Web Application accessibile da browser.	Obbligatorio	Capitolato Zucchetti
RVO02	Il sistema deve supportare almeno un LLM accessibile tramite API REST.	Obbligatorio	UC15 Capitolato Zucchetti
RVO03	Il sistema deve permettere l'integrazione e gestione di LLM multipli configurabili dall'utente.	Obbligatorio	UC15, UC15.1, UC17 Capitolato Zucchetti

## 5. Tracciamento dei Requisiti

Fonte	Requisito
UC1 e sottocasi	RFO01, RFO02, RFO03, RFO04, RFO05
UC2	RFD06, RQO02
UC3	RFD07, RQD06
UC4	RFO09
UC5	RFO08, RQO02
UC6	RFO10
UC7	RFO11, RQO02
UC8 e sottocasi	RFO12, RFO13, RFO14, RFO15, RFO16
UC9	RFD17, RQO02
UC10	RFD18, RQD06
UC11	RFP19
UC12	RFO20, RQO02
UC13	RFO21
UC14	RFO22, RQO02
UC15 e sottocasi	RFO23, RFP25, RVO02, RVO03
UC16	RFO24, RQO02
UC17 e sottocasi	RFO27, RFO28, RFO29, RVO03
UC18 e sottocasi	RFO30, RFO31, RFO32, RQP05
UC19	RFP26, RQO02
UC20	RFO33, RQP05
UC21 e sottocasi	RFO34, RFO35, RFO36, RQD06
UC22	RFO37, RQD03
UC23	RFP38, RQD03, RQP04
UC24	RFP39, RQP04
UC25	RFO40, RQO02
UC26 e sottocasi	RFO41, RFO42
UC27 e sottocasi	RFD43, RQD06
UC28	RFP44
UC29 e sottocasi	RFP44, RFO45
Capitolato Zucchetti	RVO01, RFO30, RFO33, RFO34, RFO37, RFP38, RFP39, RFO41, RFP44

### 5.1. Riepilogo

Tipologia	Obbligatorio	Opzionale	Desiderabile	Totale
Funzionali	32	8	5	45
Qualità	3	2	2	7
Vincolo	3	0	0	3
				55

## 5.2. Conclusioni

I requisiti del prodotto potranno essere modificati nel corso del suo ciclo di vita per introdurre miglioramenti o aggiornamenti. Qualora siano disponibili risorse sufficienti, il gruppo valuterà l'inserimento di nuovi requisiti volti a incrementare il valore e la qualità del prodotto. Eventuali aggiornamenti saranno quindi presi in considerazione nel tempo, in un'ottica di evoluzione continua.