

Manuale Utente

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	Giada Rossi
		Mattia Piva	Marco Cola

Registro delle modifiche

Versione	Data	Autori	Verificatori	Descrizione
v1.0.0	2025-08-24	Marco Cola	Ruize Lin	Approvazione per PB
v0.1.2	2025-08-22	Marco Cola, Ruize Lin	Giada Rossi	Aggiornate le immagini del MVP
v0.1.1	2025-08-21	Marco Cola	Ruize Lin	Correzione grammatica documento e impostazione delle pagine
v0.1.0	2025-08-18	Marco Cola	Ruize Lin, Giada Rossi	Aggiunti riferimenti al Glossario v2.0.0
v0.0.5	2025-08-11	Ruize Lin	Marco Cola, Giada Rossi	Aggiunto sezione «Guida per l'avvio dell'analisi della webapp»
v0.0.4	2025-08-06	Giulia Hu	Marco Cola, Ruize Lin	Aggiunto sezione «Supporto Tecnico», indice delle immagini
v0.0.3	2025-08-06	Ruize Lin	Marco Cola	Aggiunto Sezione «Guida per l'avvio della webapp»
v0.0.2	2025-08-05	Stefano Dal Poz	Ruize Lin, Marco Cola	Capitolo Introduzione
v0.0.1	2025-07-26	Marco Cola	Giada Rossi	Prima Bozza

Indice

1. Introduzione	5
1.1. Scopo del prodotto	5
1.2. Scopo del documento	5
1.3. Glossario	5
1.4. Riferimenti	5
1.4.1. Riferimenti normativi	5
1.4.2. Riferimenti informativi	6
2. Guida per l'avvio e uso della webapp	7
2.1. Requisiti	7
2.2. File di configurazione del progetto	7
2.3. Avvio della webapp	7
2.4. Chiusura della webapp	8
2.5. Accesso alla webapp	8
2.6. Guida all'uso	8
2.6.1. Pagina Home	8
2.6.2. Pagina Configurazione API	9
2.6.2.1. Creazione di preset	9
2.6.2.2. Modifica o eliminazione di un preset	10
2.6.3. Pagina Gestione Domande	10
2.6.3.1. Creazione di domanda/risposta attesa	11
2.6.3.2. Modifica o eliminazione di una coppia domanda/risposta attesa	11
2.6.3.3. Importazione di domanda/risposta attesa	11
2.6.4. Pagina Gestione Set di Domande	12
2.6.4.1. Creazione di set	12
2.6.4.2. Modifica o eliminazione di un set	12
2.6.4.3. Importazione set di domande	13
2.6.5. Pagina Esecuzione Test	13
2.6.5.1. Esecuzione test automatico per LLM	14
2.6.6. Pagina Visualizzazione Risultati	14
2.6.6.1. Visualizzazione risultati test	15
3. Guida per l'avvio dell'analisi della webapp	16
3.1. Installazione delle librerie necessarie per l'analisi	16
3.2. Analisi statica	16
3.2.1. Controllo di stile e linting con Flake8	16
3.2.2. Analisi dei tipi	16
3.3. Analisi dinamica	17
3.3.1. Esecuzione della suite di test con pytest	17
4. Supporto tecnico	18

Indice delle Immagini

Figura 1	Pagina Home	8
Figura 2	Pagina Configurazione API	9
Figura 3	Interfaccia Preset	10
Figura 4	Pagina Preset Salvato	10
Figura 5	Pagina Gestione Domande	11
Figura 6	Pagina Importazione Domande	12
Figura 7	Pagina Importazione Set	13
Figura 8	Pagina Esecuzione Test	13
Figura 9	Risultati riassuntivi	14
Figura 10	Pagina Visualizzazione risultati	15
Figura 11	Grafico radar tridimensionale	15

1. Introduzione


1.1. Scopo del prodotto

Lo scopo principale del *prodotto_G* è quello di fornire all'azienda *proponente_G*, Zucchetti S.p.A., una piattaforma *software_G* denominata Artificial QI, pensata per supportare la valutazione delle prestazioni di sistemi basati su Large Language Models (LLM). Il *sistema_G* nasce dall'esigenza di disporre di uno strumento centralizzato, automatizzato e configurabile che consenta di testare, analizzare e confrontare le risposte generate da diversi modelli *LLM_G* rispetto a risposte attese predefinite.

La piattaforma dovrà permettere all'utente sviluppatore di inserire e gestire domande e risposte attese, interfacciarsi con *LLM_G* esterni tramite chiamate *API_G* conformi allo standard *OpenAPI 3.1_G*, ed eseguire valutazioni automatiche della coerenza tra input e output tramite metriche personalizzabili (come *BM25_G* o modelli neurali). Inoltre, dovrà offrire *funzionalità_G* avanzate di visualizzazione, ordinamento e confronto dei risultati ottenuti, favorendo anche l'archiviazione e l'analisi storica delle esecuzioni.

Il *prodotto_G* finale sarà un'unica *applicazione_G* integrata, accessibile via web, che racchiude tutte le *funzionalità_G* necessarie in un flusso di lavoro continuo e organico. Tale soluzione dovrà garantire semplicità d'uso, flessibilità nella configurazione e supporto per l'analisi evolutiva delle prestazioni dei modelli, diventando così un valido supporto nelle attività di test e selezione degli *LLM_G* più adatti per specifici scenari applicativi.

1.2. Scopo del documento

Questo documento ha lo scopo di spiegare ai committenti le modalità di utilizzo e di verifica delle funzionalità del sistema informatico che il gruppo  7Commits ha dovuto sviluppare per adempiere alle richieste fatte in merito allo studio di un sistema per la valutazione delle prestazioni di sistemi basati su Large Language Models (*LLM_G*). Al suo interno verranno illustrate tutte le istruzioni per avviare la *webapp_G*, le istruzioni per collegarsi ad essa, una breve guida all'uso del prodotto per dimostrarne il corretto funzionamento e le *funzionalità_G* che esso espone, infine, come avviare gli stress test richiesti dal committente così da poterli modificare a piacimento, con il fine di testare più approfonditamente o in modo diverso alcune parti.

1.3. 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_G*.

1.4. Riferimenti

1.4.1. Riferimenti normativi

- **Norme di progetto v2.0.0**
- **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>
- **PD1 - Regolamento del progetto didattico**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/PD1.pdf>

1.4.2. Riferimenti informativi

- **T2 - Processi di ciclo di vita del software:**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T02.pdf>
- **T4 - Gestione di progetto:**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T04.pdf>
- **Glossario v2.0.0:**
<https://7commits.github.io/7Commits/glossary.html>

2. Guida per l'avvio e uso della webapp

2.1. Requisiti

Ci sono due modi per avviare la *webapp*: uno tramite *Python* installato localmente, l'altro tramite *Docker*. Perciò, oltre a *Git*, che deve essere installato in ogni caso, i requisiti per il primo metodo sono l'installazione di almeno Python 3.10 e di *MySQL* in versione non precedente alla 5.7. Per il secondo metodo, è necessario che siano installati sia Docker che *Docker Compose*. Tutti i requisiti sono specificati nel file *requirements.txt* all'interno del repository.

2.2. File di configurazione del progetto

Il progetto prevede l'utilizzo di alcuni file di configurazione fondamentali per l'esecuzione e lo sviluppo:

- *db.config.example*: contiene un esempio di configurazione per la connessione al database *MySQL*. Deve essere copiato, rinominato in *db.config* e adattato ai parametri reali dell'ambiente di lavoro.
- *docker-compose.yml*: definisce i servizi *Docker* necessari per l'avvio dell'applicazione e del *database*. Consente di predisporre rapidamente l'ambiente di esecuzione tramite il comando *docker-compose up --build*.
- *requirements.txt*: elenca le dipendenze indispensabili per l'esecuzione dell'applicazione (ad esempio: *Streamlit*, *pandas*, *plotly*, *openai*, *SQLAlchemy*, *PyMySQL*, *cryptography*).
- *requirements-dev.txt*: raccoglie le dipendenze dedicate allo sviluppo, comprendenti strumenti per il controllo di qualità del codice (*Flake8* per il linting, *MyPy* per il type checking) e le librerie necessarie per le attività di testing.

2.3. Avvio della webapp

Il primo passo consiste nello scaricare la webapp tramite il comando:

```
1 git clone https://github.com/7Commits/MVP
```

bash

Successivamente, accedere alla cartella **MVP** e creare un file denominato *db.config*, copiando al suo interno il contenuto di *db.config.example*. Qualora si desideri utilizzare *Python* e *MySQL* anziché *Docker*, sarà necessario modificare il file *db.config* inserendo i parametri di connessione al proprio *database* MySQL.

A questo punto, installare i pacchetti *Python* richiesti tramite il comando:

```
1 pip install -r requirements.txt
```

bash

Dopo aver verificato che la configurazione sia corretta ed aver avviato *MySQL*, è possibile lanciare la *webapp* con il comando:

```
1 python -m streamlit run app.py --server.port 8501
```

bash

In alternativa, se si preferisce utilizzare *Docker*, è sufficiente seguire le istruzioni riportate all'interno di *db.config.example* per generare correttamente il file *db.config*, quindi eseguire:

```
1 docker-compose up -d
```

bash

Al termine del processo, la *webapp* sarà avviata correttamente.

2.4. Chiusura della webapp

L'arresto dell'applicazione può essere effettuato in due modalità, a seconda della configurazione utilizzata:

- **Utilizzo con Docker:** Per interrompere l'esecuzione della webapp, eseguire il seguente comando: `bash`

```
1 docker compose stop
```

Questo comando interrompe l'esecuzione dei container e mette in pausa la webapp.

- **Utilizzo con Python + MySQL:** Nel caso in cui l'applicazione sia stata avviata tramite Python e MySQL, è sufficiente chiudere il terminale da cui è stato eseguito il comando di avvio. In questo modo il processo viene interrotto e la webapp si arresta automaticamente.

2.5. Accesso alla webapp

Una volta completata la configurazione dei file necessari e avviata la `webapp`, l'accesso all'interfaccia avviene tramite un browser compatibile con i requisiti minimi. È sufficiente aprire il browser e navigare all'indirizzo:

```
1 http://localhost:8501
```

2.6. Guida all'uso

2.6.1. Pagina Home

Dopo l'accesso al sito, viene visualizzata la pagina Home della `webapp`. Questa sezione presenta una breve descrizione dell'applicazione e offre una panoramica delle principali funzionalità disponibili. Le diverse sezioni della webapp sono facilmente raggiungibili tramite la barra di navigazione posizionata sulla sinistra dell'interfaccia.

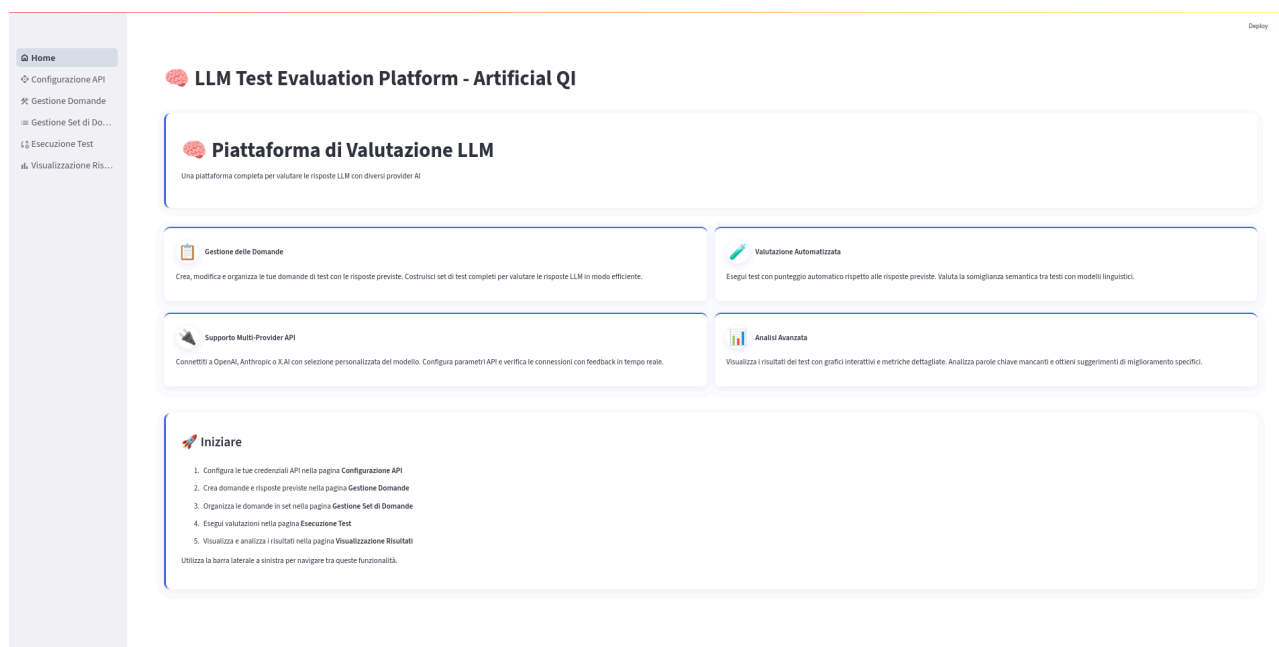


Figura 1: Pagina Home

2.6.2. Pagina Configurazione API

Selezionando la voce «Configurazione API» dalla barra di navigazione, l'utente può creare oppure modificare dei *preset*_G, ovvero configurazioni salvate che consentono al sistema di essere riutilizzate rapidamente senza doverle reinserire manualmente ogni volta.

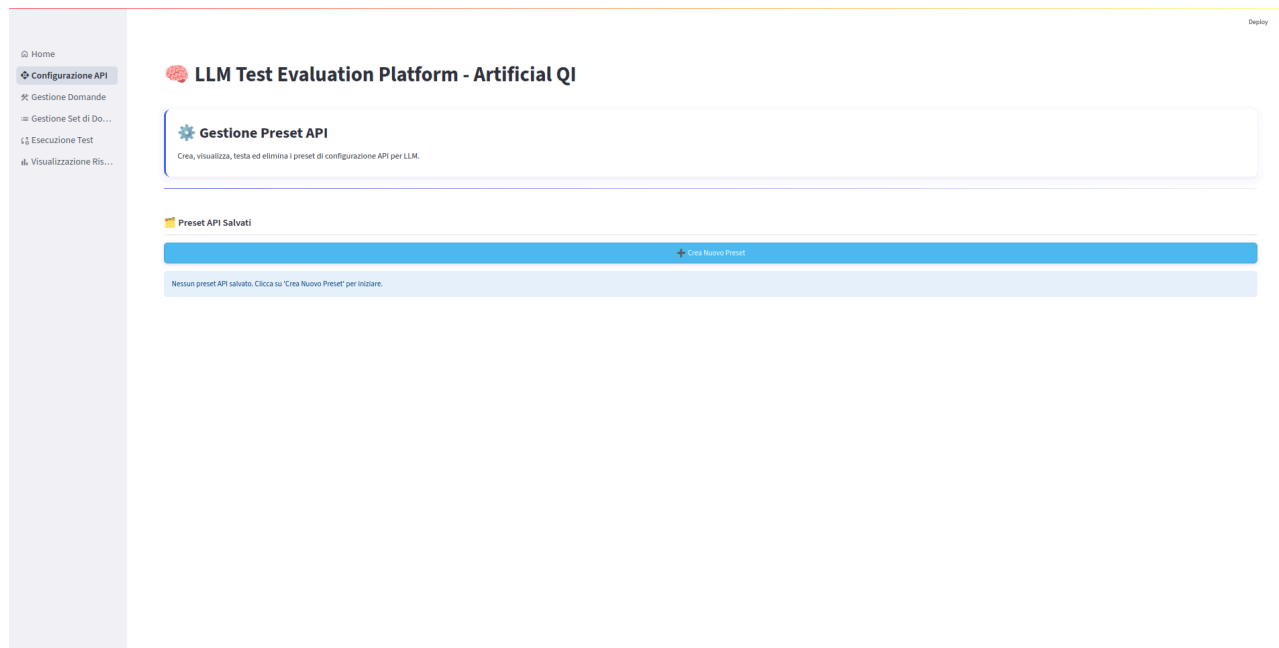


Figura 2: Pagina Configurazione API

2.6.2.1. Creazione di preset

Cliccando su “Crea nuovo preset” si aprirà l’interfaccia per la creazione di un nuovo *preset*_G.

All’interno della schermata sono presenti diversi campi da compilare:

- **Nome del Preset:** rappresenta il nome identificativo assegnato al preset.
- **Chiave API:** la chiave di accesso fornita dalla piattaforma LLM.
- **Provider Endpoint:** l'*URL*_G sul quale la *webapp*_G invierà le richieste al *modello* *LLM*_G.
- **Modello:** il modello *LLM*_G che si intende utilizzare.
- **Temperatura:** parametro che controlla il grado di casualità delle risposte generate; valori più alti producono risposte maggiormente variabili e creative.

LLM Test Evaluation Platform - Artificial QI

Gestione Preset API

Crea, visualizza, testa ed elimina i preset di configurazione API per LLM.

Modifica/Crea Preset API

Nome del Preset

Chiave API

Provider Endpoint
http://api.openai.com/v1

Modello
gpt-4o

Temperatura
1.0

Max Tokens
1000

Testa Connessione API

Salva Preset

Modifica

Elimina

Figura 3: Interfaccia Preset

Se la compilazione dei campi è avvenuta correttamente, premendo il pulsante «Salva preset» la configurazione verrà registrata e resa disponibile per utilizzi successivi.

2.6.2.2. Modifica o eliminazione di un preset

Dopo aver salvato almeno un *preset*, questo sarà sempre disponibile all'interno della sezione «Configurazione API». Da qui, l'utente potrà scegliere se modificare i parametri precedentemente inseriti oppure eliminare definitivamente il preset.

LLM Test Evaluation Platform - Artificial QI

Gestione Preset API

Crea, visualizza, testa ed elimina i preset di configurazione API per LLM.

Preset API Salvati

+ Crea Nuovo Preset

Nebius

Modello: gpt-4o

Endpoint: http://api.studio.nebius.com/v1

Modifica

Elimina

Figura 4: Pagina Preset Salvato

2.6.3. Pagina Gestione Domande

Nella pagina «Gestione Domande» l'utente può amministrare le coppie composte da domanda e risposta attesa, che costituiscono la base per la creazione e l'esecuzione dei test.

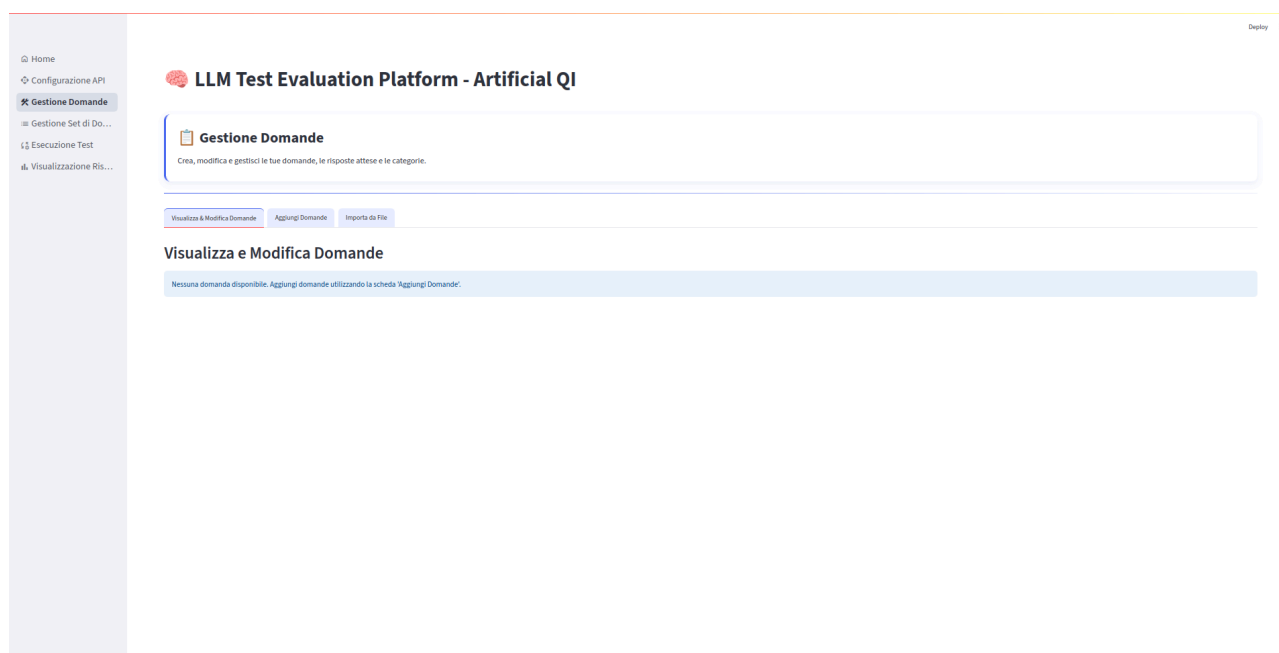


Figura 5: Pagina Gestione Domande

2.6.3.1. Creazione di domanda/risposta attesa

Cliccando su «Aggiungi Domande» si apre l'interfaccia per la creazione di una nuova coppia domanda/risposta attesa.

All'interno del form sono presenti i seguenti campi da compilare:

- **Domanda:** rappresenta l'input che verrà sottoposto al modello LLM per ottenere una risposta.
- **Risposta attesa:** indica il contenuto che il modello dovrebbe restituire, ossia la struttura o la formulazione ritenuta corretta come risposta ideale.
- **Categoria (opzionale):** consente di organizzare le domande in gruppi tematici, facilitando la classificazione e la ricerca all'interno del sistema.

2.6.3.2. Modifica o eliminazione di una coppia domanda/risposta attesa

Una volta salvata almeno una coppia domanda/risposta attesa, questa sarà visibile all'interno della sezione «Visualizza e Modifica Domande». Da qui, l'utente potrà scegliere se modificare o eliminare una domanda.

Per modificare una voce è sufficiente selezionarla dall'elenco, aggiornare i campi desiderati e premere il pulsante «Salva modifiche» per confermare le variazioni. In alternativa, è possibile rimuovere definitivamente una domanda selezionando l'opzione «Elimina domanda».

2.6.3.3. Importazione di domanda/risposta attesa

Cliccando su «Importa da file» si apre l'interfaccia dedicata all'importazione di una o più coppie domanda/risposta attesa. L'importazione può avvenire tramite un file in formato [CSV_G](#) oppure [JSON_G](#), a condizione che il file rispetti la struttura prevista. A supporto dell'utente sono forniti esempi che illustrano il formato corretto dei file accettati.

Dopo aver selezionato il file desiderato, comparirà il pulsante «Importa Domande». Premendolo, tutte le coppie valide contenute nel file verranno caricate automaticamente all'interno del sistema.

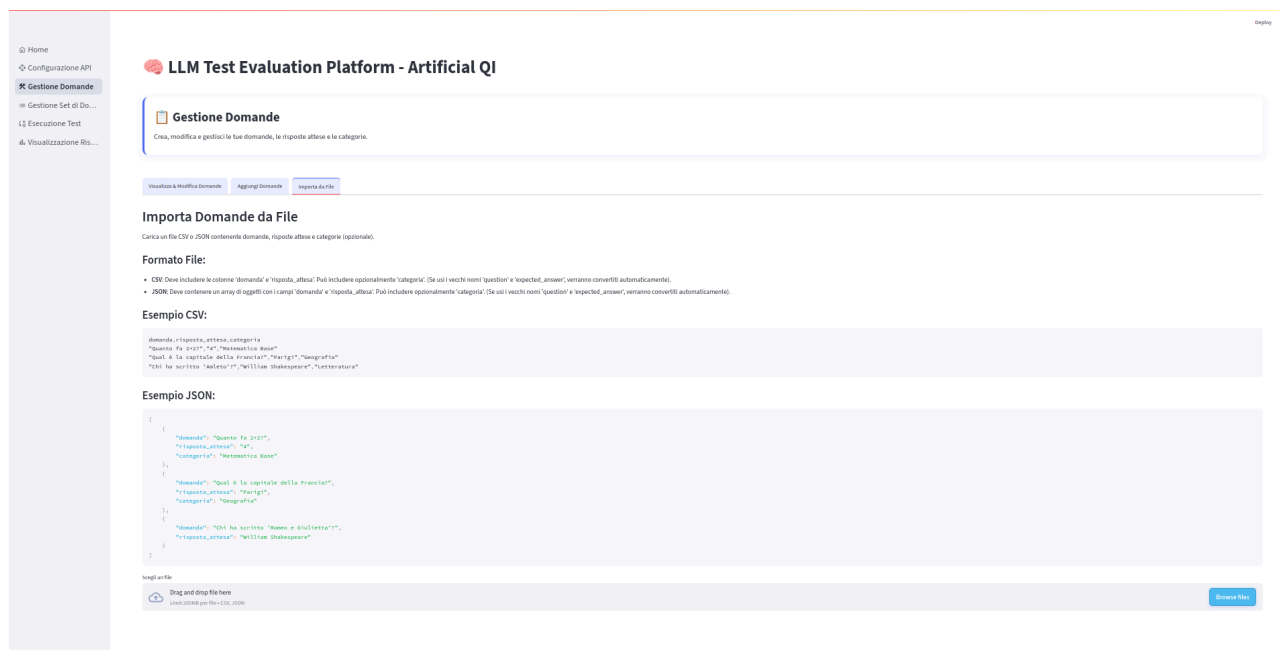


Figura 6: Pagina Importazione Domande

2.6.4. Pagina Gestione Set di Domande

Nella pagina «Gestione Set di Domande» l'utente può creare insiemi composti da differenti tipologie di domande. Ogni set rappresenta un raggruppamento di più domande, al quale è possibile assegnare un nome identificativo per facilitarne la gestione e il successivo utilizzo.

2.6.4.1. Creazione di set

Cliccando su «Crea Nuovo Set» si apre l'interfaccia dedicata alla creazione di un nuovo set di domande.

All'interno della schermata sono presenti i seguenti campi e opzioni:

- **Nome Set:** il nome identificativo da assegnare al set.
- **Selezione delle domande:** cliccando sulla barra sotto la voce «Seleziona domande per questo set» si apre un menu a tendina contenente l'elenco di tutte le domande salvate nel database. Da qui è possibile scegliere le domande da includere nel set.

Una volta completata la selezione, è sufficiente premere il pulsante «Crea Set» per confermare l'operazione e rendere disponibile il nuovo set all'interno del sistema.

2.6.4.2. Modifica o eliminazione di un set

Una volta salvato almeno un set di domande, questo sarà sempre disponibile nella sezione «Visualizza & Modifica Set». Da qui l'utente può scegliere se modificare o eliminare un set esistente.

Per modificare un set è sufficiente cliccare sul nome del set desiderato: la sezione corrispondente si espanderà, permettendo di aggiungere o rimuovere domande tramite selezione o deselection, nonché di aggiornare il nome del set. Al termine delle modifiche, è necessario premere il pulsante «Salva Modifiche» per confermare.

In alternativa, è possibile eliminare definitivamente un set selezionando l'opzione «Elimina Set». In questo caso viene richiesta una riconferma dell'operazione, al fine di evitare cancellazioni accidentali.

2.6.4.3. Importazione set di domande

Cliccando su “Importa Set da file” si aprirà l’interfaccia per l’importazione di un o più set di coppie di domanda/risposta attesa, tramite un file CSV oppure JSON che segua una struttura precisa. Sono forniti esempi su come devono essere strutturati i vari file. Dopo aver selezionato il file, apparirà il pulsante “Importa Set”. Premendolo, verranno importate tutti i set, purché il file rispetti la struttura richiesta.

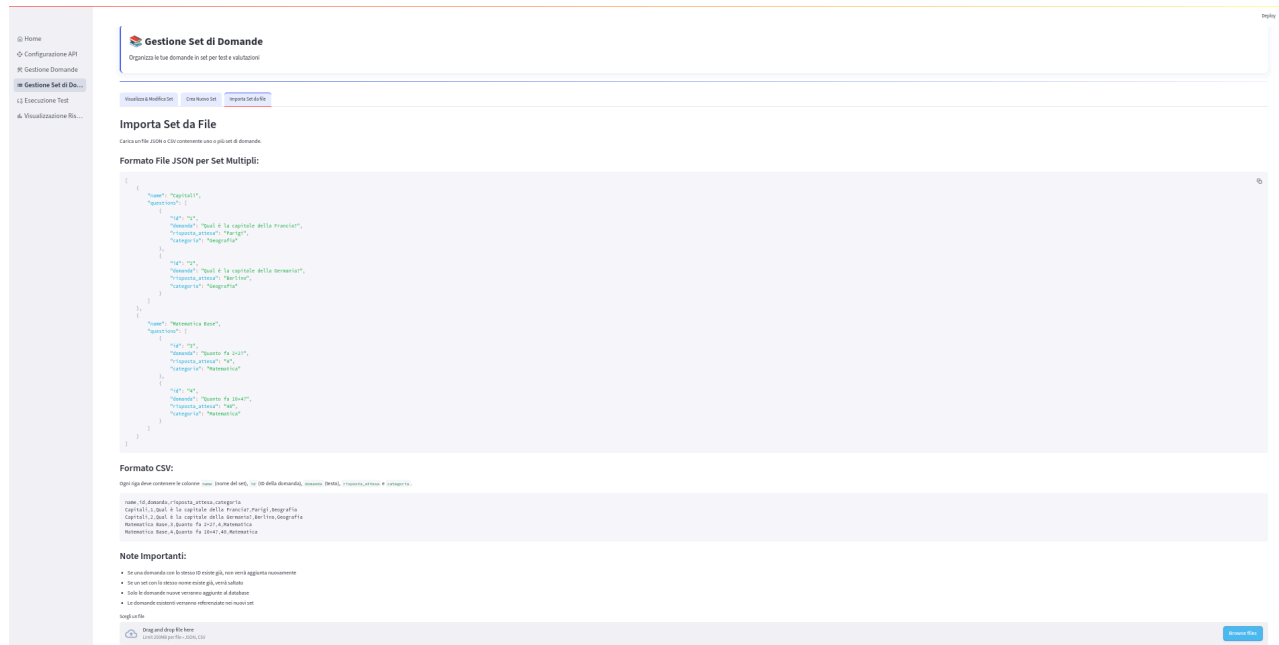


Figura 7: Pagina Importazione Set

2.6.5. Pagina Esecuzione Test

Nella pagina «Esecuzione Test» l’utente può avviare test finalizzati a valutare le prestazioni di un modello LLM_G nel rispondere a specifiche tipologie di domande. La valutazione si basa su tre metriche principali: **somiglianza**, **correttezza** e **completezza**, calcolate confrontando le risposte generate con quelle attese.

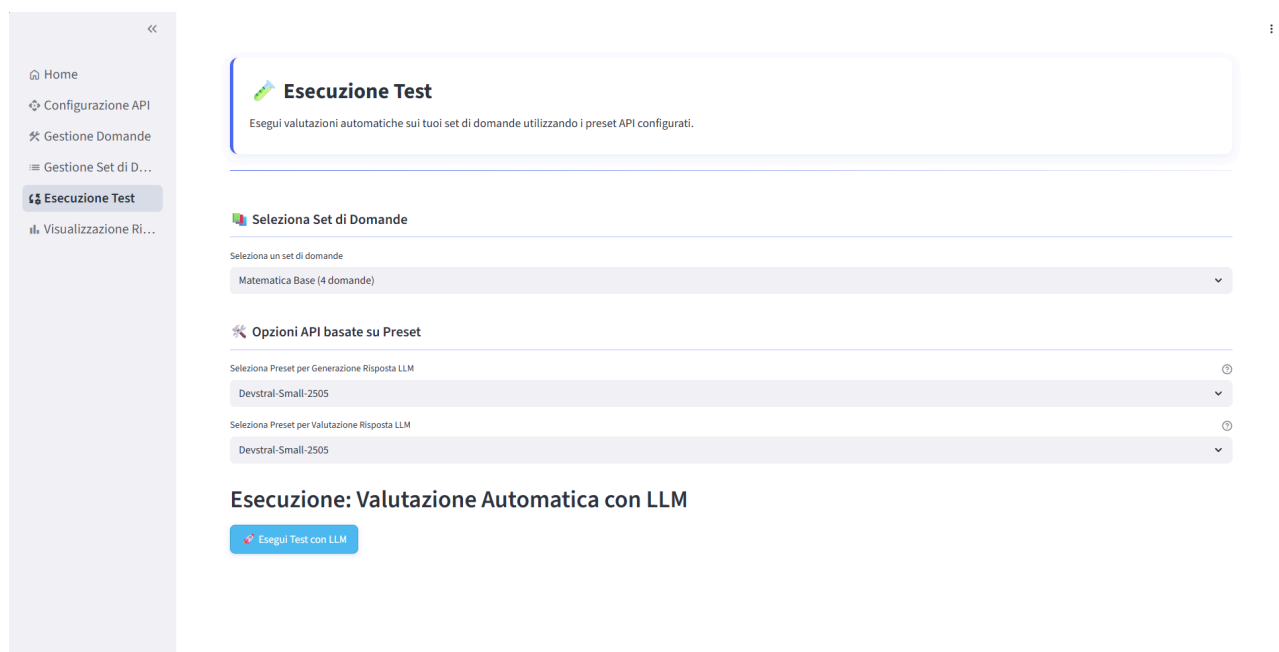


Figura 8: Pagina Esecuzione Test

2.6.5.1. Esecuzione test automatico per LLM

Per poter eseguire i test è necessario, in primo luogo, aver configurato almeno un [preset_G](#) nella pagina «Configurazione API» e aver creato un set di domande nella pagina «Gestione Set di Domande».

Successivamente, l'utente può procedere selezionando:

- il **set di domande** da utilizzare per il test;
- il **preset** con cui generare le risposte del modello LLM;
- il **preset** con cui valutare la qualità delle risposte prodotte.

Completata la selezione, è sufficiente cliccare sul pulsante «Esegui Test con LLM» per avviare la valutazione automatica.

Al termine del processo, i **risultati riassuntivi** vengono visualizzati direttamente nella stessa pagina, consentendo un'immediata analisi delle prestazioni del modello.

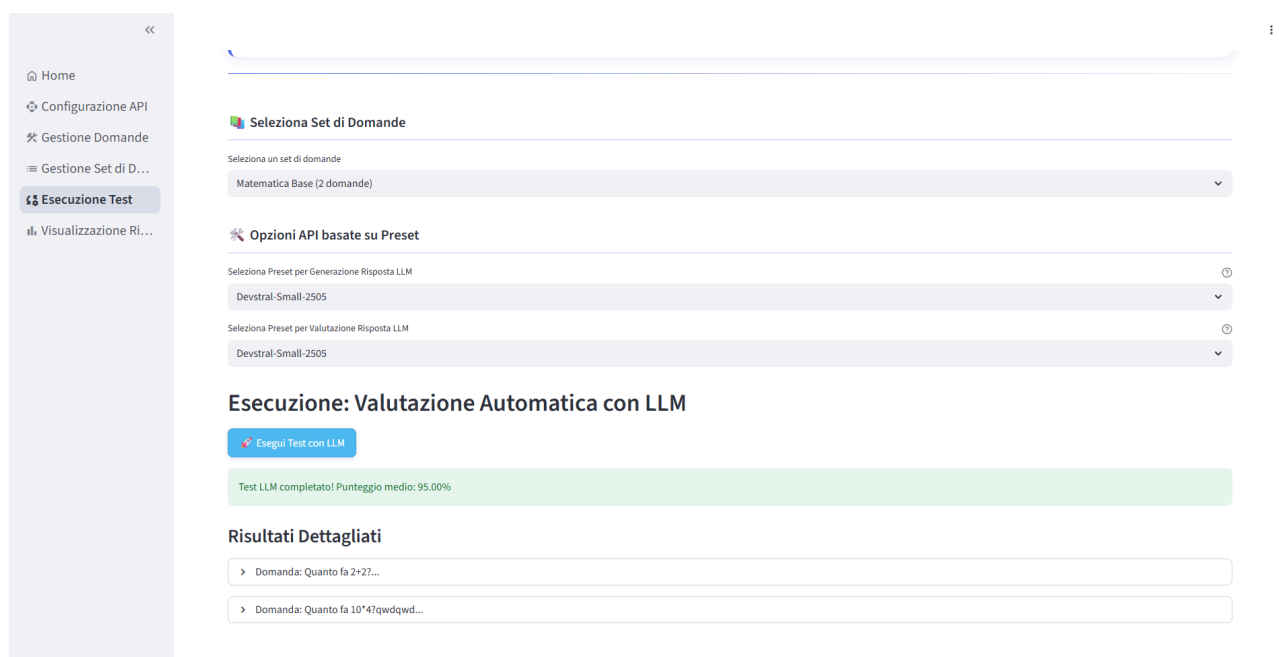


Figura 9: Risultati riassuntivi

2.6.6. Pagina Visualizzazione Risultati

Nella pagina “Esecuzione Test” l'utente ha la possibilità di consultare i risultati dettagliati di ogni valutazione oppure di confrontarli con quelli di altri test, al fine di osservare le differenze nelle prestazioni tra diversi [modelli LLM_G](#) sul medesimo set di domande.

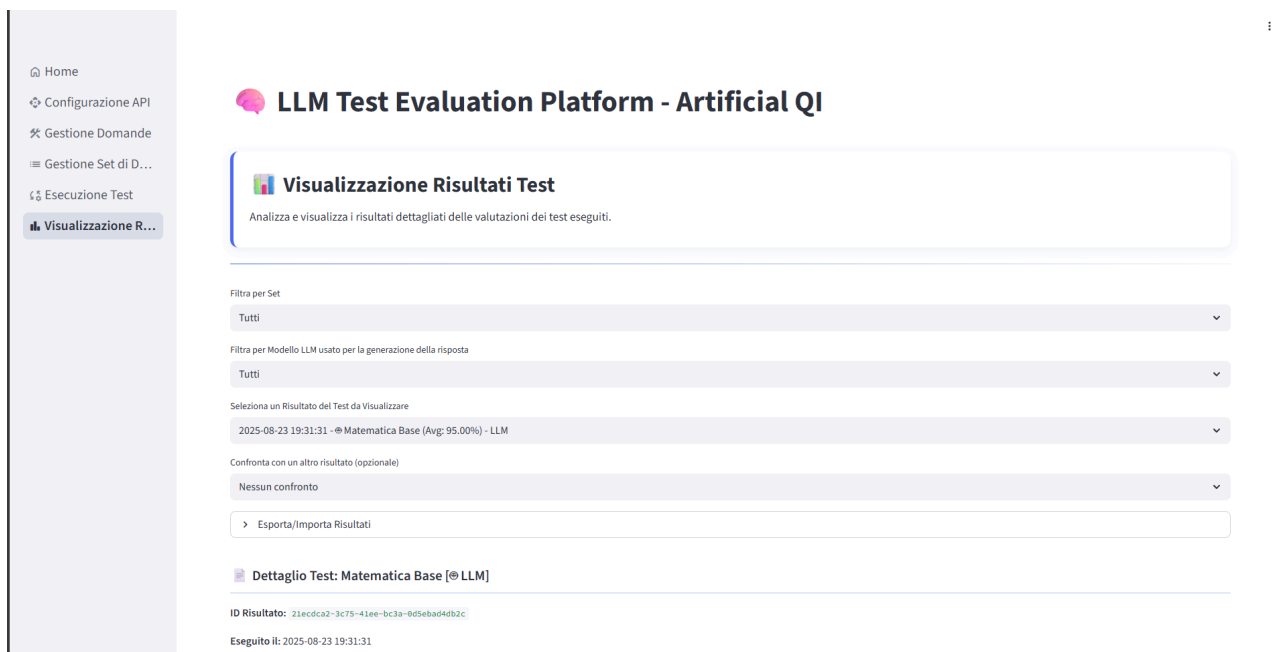


Figura 10: Pagina Visualizzazione risultati

2.6.6.1. Visualizzazione risultati test

È possibile ricercare rapidamente i risultati applicando appositi filtri, ad esempio in base al set di domande utilizzato o al *modello LLM_G* impiegato nella generazione delle risposte.

Una volta selezionato un risultato, la pagina visualizza:

- un **grafico radar tridimensionale** che rappresenta le tre metriche di valutazione (*Somiglianza*, *Correttezza* e *Completezza*) calcolate sull'intero set di domande;
- i **punteggi associati a ciascuna domanda**, ottenuti come media delle tre metriche.

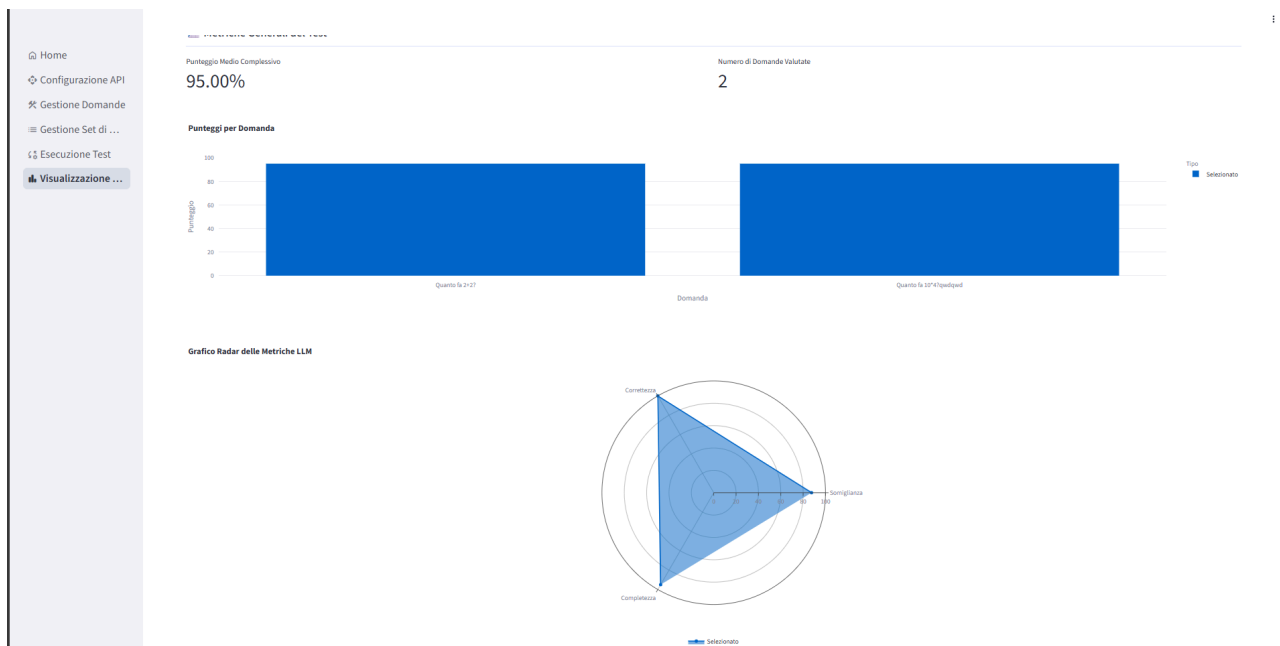


Figura 11: Grafico radar tridimensionale

3. Guida per l'avvio dell'analisi della webapp

3.1. Installazione delle librerie necessarie per l'analisi

L'installazione delle librerie richieste dipende dal metodo utilizzato per l'esecuzione dell'applicazione:

- **Esecuzione con Python + MySQL:**

È possibile installare direttamente le librerie necessarie tramite il comando:

```
1 pip install -r requirements-dev.txt
```

bash

Questo comando provvede a installare gli strumenti indispensabili per l'analisi della webapp.

- **Esecuzione con Docker:**

In questo caso, è necessario accedere al container dell'applicazione con il comando:

```
1 docker exec -it llm-app /bin/bash
```

bash

Una volta entrati nel container, si procede con lo stesso comando di installazione:

```
1 pip install -r requirements-dev.txt
```

bash

Si ricorda che, ad ogni nuova ricostruzione del container, questa procedura deve essere ripetuta.

3.2. Analisi statica

L'*analisi statica*_G consiste nell'esame del codice sorgente senza eseguirlo, con l'obiettivo di individuare *bug*_G, violazioni delle convenzioni di stile, problemi di tipizzazione o potenziali vulnerabilità.

Per la nostra *webapp*_G sono già stati predisposti due strumenti principali:

- *Flake8*_G: per il controllo della conformità del codice agli standard di stile *Python*_G
- *MyPy*_G: per la verifica della correttezza della tipizzazione statica.

3.2.1. Controllo di stile e linting con Flake8

*Flake8*_G verifica la conformità del codice alle *convenzioni PEP 8*_G e segnala eventuali errori comuni, inclusi problemi di sintassi. Le regole di controllo possono essere personalizzate tramite il file di configurazione `.flake8` presente nel progetto.

Per eseguire il controllo è sufficiente utilizzare il comando:

```
1 python -m flake8 app.py controllers/ models/ views/ utils/
```

bash

Eventuali errori o avvisi rilevati verranno mostrati direttamente nel terminale, consentendo una rapida individuazione e correzione.

3.2.2. Analisi dei tipi

*MyPy*_G esegue un controllo statico dei tipi basato sulle annotazioni presenti nel codice. Lo strumento rileva eventuali incongruenze tra i tipi dichiarati e quelli effettivamente utilizzati (ad esempio, il tentativo di sommare una stringa con un numero intero).

Per avviare l'analisi è sufficiente eseguire il comando:

```
1 python -m mypy app.py controllers/ models/ views/ utils/
```

bash

Analogamente a quanto avviene con Flake8, eventuali errori o avvisi verranno mostrati direttamente nel terminale, consentendo una rapida individuazione dei problemi.

3.3. Analisi dinamica

L'analisi dinamica consiste nell'esecuzione del codice per osservarne il comportamento a runtime.

Questa tecnica permette di:

- verificare che il programma funzioni correttamente in diversi scenari di utilizzo;
- individuare eventuali malfunzionamenti o anomalie non rilevabili con l'analisi statica;
- misurare la copertura dei test, ossia la percentuale di codice effettivamente eseguita;
- valutare le prestazioni e identificare possibili colli di bottiglia.

3.3.1. Esecuzione della suite di test con pytest

Il `repository` include una cartella `tests` contenente i file di test, accompagnata da un file `pytest.ini` che specifica la directory di riferimento. I test implementati coprono diverse funzioni di logica applicativa, come ad esempio:

- il calcolo delle statistiche sui risultati,
- l'importazione e la gestione dei set di domande.

Per eseguire la suite di test è sufficiente utilizzare il comando:

```
1  pytest --cov=. -vv
```

bash

Al termine dell'esecuzione, nel terminale verrà riportato l'esito di ciascun test, con indicazione chiara di quelli superati e di eventuali fallimenti.

4. Supporto tecnico

Il nostro team è a disposizione per fornire supporto in caso di dubbi o difficoltà relative all'utilizzo del prodotto.

Per richieste di assistenza o chiarimenti, è possibile contattarci all'indirizzo email 7commits@gmail.com, avendo cura di includere una descrizione dettagliata del problema riscontrato ed eventuali messaggi di errore ricevuti.

Faremo il possibile per analizzare tempestivamente la segnalazione, risolvere l'inconveniente e fornire tutte le informazioni necessarie a garantire la continuità d'uso dell'applicazione.