



Università Politecnica delle Marche

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria Informatica e
dell'Automazione

CORSO DI DATA SCIENCE

**AI Generativa per la realizzazione di un software e la
generazione di dashboard**

Professore

Prof. Domenico Ursino

Studenti

Matteo Risolo

Niccolò De Pascali

ANNO ACCADEMICO 2025/2026

Indice

1	Introduzione	5
2	Generazione di codice per lo sviluppo di un software	7
2.1	Requisiti funzionali	7
2.2	Esperimento: Generazione di Applicazione GUI	8
2.3	Prima versione	8
2.4	Seconda versione	10
2.5	Terza versione	12
2.6	Quarta versione	13
2.7	Quinta versione	14
2.8	Sesta versione	16
2.9	Settima versione	16
2.10	Ottava versione	18
2.11	Nona Versione	20
2.12	Decima e ultima versione	22
2.13	Conclusioni	24
3	Attività di prompt engineering per la generazione di una dashboard	26
3.1	Modello utilizzato	26
3.2	Diagramma E-R di partenza	27
3.3	Dashboard per la tabella Utenti	27
3.3.1	Primo prompt	27

3.3.2	Secondo prompt	30
3.3.3	Terzo prompt	31
3.3.4	Quarto prompt	32
3.3.5	Quinto prompt	33
3.3.6	Sesto prompt	34
3.4	Dashboard per la tabella Libri	35
3.4.1	Primo prompt	35
3.4.2	Secondo prompt	36
3.5	Dashboard per la tabella Prenotazioni	37
3.5.1	Primo prompt	37
3.5.2	Secondo prompt	38
3.5.3	Terzo prompt	39
3.5.4	Quarto prompt	40
3.6	Conclusioni	41

Capitolo 1

Introduzione

Negli ultimi anni l'**Intelligenza Artificiale (AI)** ha conosciuto un'evoluzione significativa, affermandosi come uno degli ambiti di ricerca e applicazione più rilevanti nel contesto dell'informatica e della Data Science. In particolare, l'AI generativa rappresenta una classe di tecniche e modelli capaci di produrre nuovi contenuti a partire da dati di input, quali testi, immagini, codice sorgente e rappresentazioni strutturate di informazioni.

Secondo le definizioni fornite a livello istituzionale, i sistemi di AI generativa rientrano nella più ampia categoria dei sistemi di AI basati su modelli di apprendimento automatico, in grado di apprendere pattern dai dati e di generare output coerenti e contestualmente rilevanti sulla base di istruzioni espresse in linguaggio naturale o formale. Tali sistemi trovano applicazione in numerosi ambiti, tra cui lo sviluppo software, l'analisi dei dati, la visualizzazione delle informazioni e il supporto ai processi decisionali.

Nel contesto della Data Science, l'AI generativa non si limita all'analisi statistica o predittiva dei dati, ma si configura come uno strumento di supporto alla progettazione e all'implementazione di soluzioni software data-driven. In particolare, essa consente di:

- automatizzare o assistere la generazione di codice;
- facilitare la prototipazione rapida di applicazioni;

- supportare la creazione di interfacce e dashboard per l’analisi dei dati;
- migliorare l’uso e l’interpretabilità delle informazioni attraverso rappresentazioni visuali.

Alla luce di tali considerazioni, il presente progetto di Data Science si propone di analizzare e sperimentare l’impiego dell’AI generativa attraverso due task applicativi distinti ma complementari.

Il primo task riguarda la **generazione di codice**, con riferimento allo sviluppo di un software per la risoluzione di equazioni di secondo grado. In questo contesto, l’AI generativa viene utilizzata come strumento di supporto allo sviluppo di un’applicazione che, ricevuti in input i coefficienti dell’equazione, restituisce in output le soluzioni calcolate e gestisce in modo appropriato eventuali condizioni di errore, tramite un’interfaccia grafica destinata all’utente finale.

Il secondo task è invece orientato alla **generazione di una dashboard** per l’analisi dei dati, con riferimento alla gestione dei prestiti di una biblioteca. In tale ambito, l’AI generativa supporta la progettazione di una dashboard capace di rappresentare l’andamento dei prestiti nel tempo, fornendo indicatori aggregati e visualizzazioni utili all’analisi dei libri più richiesti, dei generi maggiormente consultati e dell’attività degli utenti, sulla base di un modello dei dati strutturato secondo uno schema Entity–Relationship.

Entrambi i task consentono di evidenziare come l’AI generativa possa essere integrata in un processo di sviluppo controllato e consapevole, nel quale il ruolo umano rimane centrale nelle scelte progettuali, nella validazione dei risultati e nella verifica della correttezza logica e funzionale delle soluzioni prodotte. L’obiettivo non è la sostituzione dello sviluppatore o del data scientist, bensì l’utilizzo dell’AI generativa come strumento di supporto intelligente.

Capitolo 2

Generazione di codice per lo sviluppo di un software

Il primo task del progetto è dedicato alla generazione di codice tramite AI generativa ed è finalizzato allo sviluppo di un software per la risoluzione di equazioni di secondo grado. L'obiettivo principale è analizzare come un modello di AI generativa possa supportare le fasi di progettazione e implementazione di un'applicazione software, a partire da requisiti funzionali espressi in linguaggio naturale.

Il software oggetto del task è un risolutore di equazioni di secondo grado, in grado di operare su equazioni della forma:

$$ax^2 + bx + c = 0 \quad (2.1)$$

dove a , b e c rappresentano coefficienti reali forniti in input dall'utente.

2.1 Requisiti funzionali

Il software sviluppato deve soddisfare i seguenti requisiti funzionali:

- acquisizione dei coefficienti a , b e c tramite un'interfaccia grafica;
- calcolo del discriminante $\Delta = b^2 - 4ac$

- determinazione delle soluzioni dell’equazione in funzione del valore del discriminante;
- gestione dei casi particolari, quali:
 - coefficiente $a = 0$, che rende l’equazione non di secondo grado;
 - discriminante negativo, che comporta l’assenza di soluzioni reali;
- visualizzazione dei risultati o di opportuni messaggi di errore all’utente.

Tali requisiti consentono di verificare la capacità dell’AI generativa di produrre codice corretto non solo dal punto di vista sintattico, ma anche logico e matematico.

2.2 Esperimento: Generazione di Applicazione GUI

In questa fase del progetto, è stato testato il modello sulla generazione di un’applicazione completa con interfaccia grafica (GUI) e logica matematica integrata.

2.3 Prima versione

Il primo prompt che è stato fornito all’ AI è il seguente:

Crea un'applicazione in Python con un'interfaccia utente grafica utilizzando la libreria *tkinter* per risolvere equazioni di secondo grado del tipo $ax^2 + bx + c = 0$.

L'applicazione deve permettere all'utente di inserire i coefficienti a , b e c e di ottenere le soluzioni dell'equazione.

L'interfaccia deve avere:

tre campi di input per l'inserimento dei valori di a , b e c ;

un pulsante "Risovi" per avviare il calcolo;

un pulsante "C" (clear) per azzerare i campi di input e l'area di output;

un'area di output (display) che mostri:

il valore del discriminante $\Delta = b^2 - 4ac$;

le soluzioni dell'equazione, gestendo i casi $\Delta > 0$, $\Delta = 0$ e $\Delta < 0$.

L'applicazione deve gestire input non validi (valori non numerici) e il caso $a = 0$, segnalando che non si tratta di un'equazione di secondo grado.

Scrivi il codice completo ed eseguibile utilizzando esclusivamente la libreria *tkinter*.

Figura 2.1: Primo prompt

Il prompt fornisce all'AI una richiesta chiara e strutturata per la realizzazione completa di un'applicazione desktop in Python, utilizzando esclusivamente la libreria *tkinter*, finalizzata alla risoluzione di equazioni di secondo grado. Vengono specificati in modo esplicito sia gli aspetti funzionali (inserimento dei coefficienti, calcolo del discriminante e delle soluzioni) sia quelli interattivi, legati alla presenza di un'interfaccia grafica destinata all'utente finale. In particolare, il prompt guida l'AI nella progettazione di una GUI dotata di tre campi di input per i coefficienti, di un pulsante per l'avvio del calcolo e di un pulsante per il reset dell'interfaccia. Viene inoltre richiesta un'area di output in grado di visualizzare il valore del discriminante e le soluzioni dell'equazione, distinguendo correttamente i vari casi. Un elemento rilevante del prompt è l'attenzione alla robustezza dell'applicazione: l'AI è infatti istruita a gestire

input non validi e il caso particolare $a = 0$. Il risultato ottenuto è il seguente:

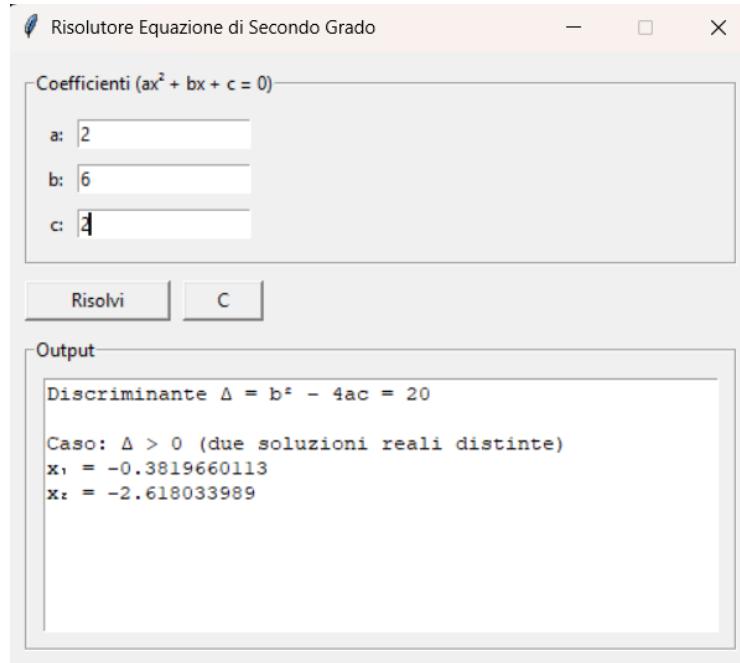


Figura 2.2: Visualizzazione del risolutore

L'output consiste in un'applicazione desktop con interfaccia grafica essenziale e coerente con il prompt. Sono presenti i campi di input per i coefficienti a , b e c , i pulsanti per l'avvio del calcolo e per il reset, e un'area di output dedicata alla visualizzazione dei risultati. L'applicazione mostra correttamente il valore del discriminante e le soluzioni dell'equazione, distinguendo il caso di discriminante positivo. Nel complesso, il risultato rispetta i requisiti funzionali richiesti e dimostra una corretta integrazione tra logica di calcolo e interfaccia grafica.

2.4 Seconda versione

Successivamente è stato dato il seguente prompt:

```

1 =====
2 Data e ora: 21/01/2026 17:46:05
3 Equazione: 2x2 + 6x + 2 = 0
4 Discriminante Δ = 20
5
6 Caso Δ > 0 (due soluzioni reali distinte)
7 x1 = -0.3819660113
8 x2 = -2.618033989
9
[H] 0

```

Figura 2.4: Visualizzazione del file di testo

Aggiungi all'applicazione per la risoluzione di equazioni di secondo grado il salvataggio automatico dei risultati in un file di testo (.txt) ogni volta che viene premuto il pulsante "Risovi".

Figura 2.3: Secondo prompt

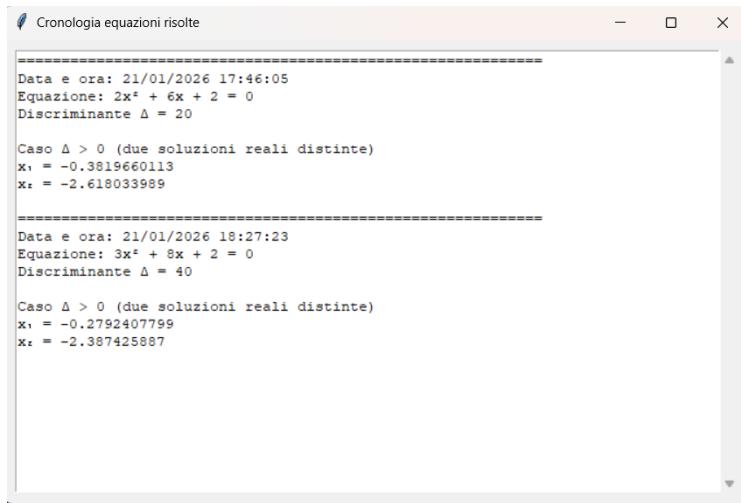
Il secondo prompt richiede l'estensione dell'applicazione già sviluppata, introducendo una funzionalità aggiuntiva di salvataggio automatico dei risultati. In particolare, viene chiesto all'AI di modificare il codice in modo che, ad ogni pressione del pulsante "Risovi", l'output dell'elaborazione venga scritto in un file di testo in formato .txt. Il prompt consente di valutare la capacità dell'AI generativa di integrare nuove funzionalità in un software esistente, intervenendo sulla logica dell'applicazione senza alterarne il comportamento principale, e di gestire operazioni di input/output su file in modo coerente con il flusso dell'interfaccia grafica. Il risultato mostra la corretta implementazione del salvataggio automatico dei dati di output in un file di testo. Ad ogni pressione del pulsante "Risovi", l'applicazione genera un file .txt contenente data e ora dell'elaborazione, l'equazione risolta, il valore del discriminante e le soluzioni calcolate, con una struttura testuale ordinata e leggibile, senza intaccare il comportamento principale.

2.5 Terza versione

Il terzo prompt richiede un’ulteriore estensione dell’applicazione, introducendo una funzionalità di consultazione della cronologia delle equazioni risolte. In particolare, viene chiesto di aggiungere all’interfaccia grafica un nuovo pulsante che consenta all’utente di visualizzare i risultati precedentemente salvati, leggendo i dati direttamente dal file di testo utilizzato per il salvataggio automatico.

Aggiungi all’interfaccia grafica un pulsante che permetta di visualizzare la cronologia delle equazioni risolte, leggendo i dati dal file di testo già utilizzato per il salvataggio automatico.

Figura 2.5: Terzo prompt



```
Cronologia equazioni risolte
=====
Data e ora: 21/01/2026 17:46:05
Equazione: 2x2 + 6x + 2 = 0
Discriminante Δ = 20

Caso Δ > 0 (due soluzioni reali distinte)
x1 = -0.3819660113
x2 = -2.618033989

=====
Data e ora: 21/01/2026 18:27:23
Equazione: 3x2 + 8x + 2 = 0
Discriminante Δ = 40

Caso Δ > 0 (due soluzioni reali distinte)
x1 = -0.2792407799
x2 = -2.387425887
```

Figura 2.6: Visualizzazione della consultazione della cronologia

Il risultato mostra l’introduzione di una finestra dedicata alla visualizzazione della cronologia delle equazioni risolte. Premendo l’apposito pulsante, l’applicazione legge il contenuto del file di testo utilizzato per il salvataggio automatico e lo visualizza in un’area testuale scorrevole. La cronologia riporta in modo ordinato la data e l’ora di ciascuna elaborazione, l’equazione risolta, il valore del discriminante e le soluzioni calcolate. L’output risulta coerente con le specifiche del prompt.

te con i dati precedentemente salvati e migliora l'usabilità dell'applicazione, consentendo all'utente di consultare lo storico delle operazioni effettuate.

2.6 Quarta versione

Il prompt successivo è il seguente:

Aggiungi all'interfaccia grafica un pulsante che permetta di cancellare la cronologia delle equazioni risolte, eliminando anche il contenuto del file di testo utilizzato per il salvataggio.

Figura 2.7: Quarto prompt

Il prompt richiede l'aggiunta di una nuova funzionalità all'interfaccia grafica, introducendo un pulsante dedicato alla cancellazione della cronologia delle equazioni risolte. In particolare, viene richiesto che l'azione di cancellazione non si limiti alla sola interfaccia, ma comporti anche l'eliminazione del contenuto del file di testo utilizzato per il salvataggio automatico dei risultati.

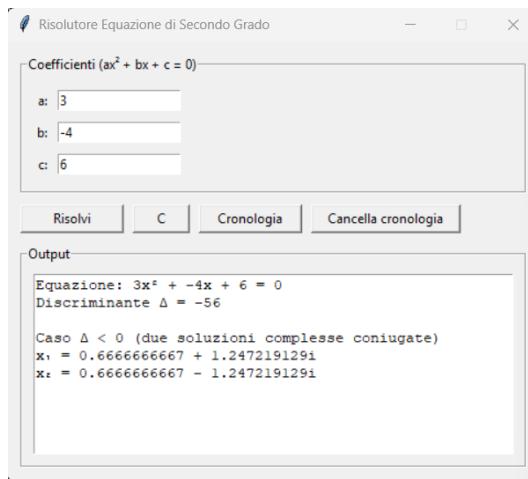


Figura 2.8: Visualizzazione del tasto per cancellare la cronologia

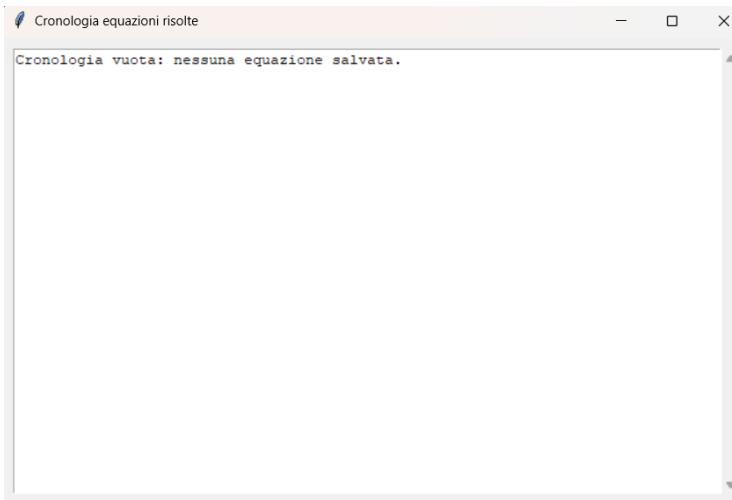


Figura 2.9: Cronologia svuotata correttamente

Il risultato conferma la corretta implementazione della funzionalità di cancellazione della cronologia. L’interfaccia grafica è stata estesa con un pulsante dedicato (“Cancella cronologia”) che consente di eliminare il contenuto del file di testo utilizzato per il salvataggio automatico.

Dopo l’operazione di cancellazione, la finestra di consultazione della cronologia mostra un messaggio esplicativo che indica l’assenza di equazioni salvate, mentre l’applicazione principale continua a funzionare regolarmente per nuove elaborazioni.

2.7 Quinta versione

Il prompt per la quinta versione è stato il seguente:

Migliora l’interfaccia grafica dell’applicazione tkinter per la risoluzione di equazioni di secondo grado, mantenendo tutte le funzionalità attuali.
Rendi l’interfaccia più chiara e ordinata migliorando layout,
allineamento dei widget, spaziature, etichette e formattazione
dell’output, senza introdurre nuove funzionalità logiche.

Figura 2.10: Quinto prompt

Il quinto prompt richiede un intervento mirato sul solo aspetto grafico e organizzativo dell'applicazione, senza introdurre nuove funzionalità logiche. L'obiettivo è migliorare la chiarezza e l'usabilità dell'interfaccia tkinter, intervenendo su layout, allineamento dei widget, spaziature, etichette e formattazione dell'output.

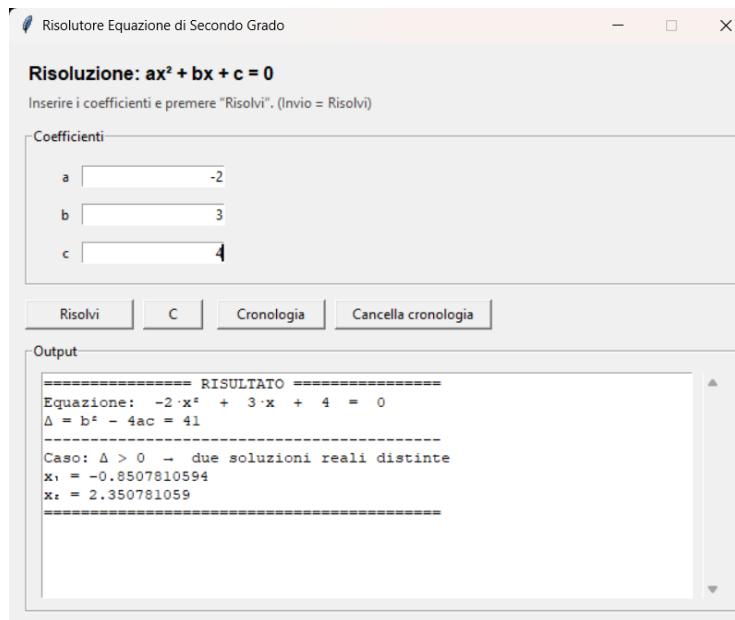


Figura 2.11: Visualizzazione dell'interfaccia migliorata

Il risultato mostra un miglioramento complessivo dell'interfaccia grafica, mantenendo inalterate tutte le funzionalità già presenti. L'applicazione presenta ora una struttura più ordinata, con una chiara separazione tra area di input, pulsanti di controllo e area di output.

Il layout risulta più leggibile grazie a un migliore allineamento dei campi, a spaziature più uniformi e a etichette più esplicative. Anche l'output è formato in modo più chiaro, con intestazioni e separatori che facilitano la lettura dei risultati.

Modifica la gestione della cronologia in modo che solo le equazioni risolte correttamente vengano salvate nel file di testo, escludendo i messaggi di errore.

Figura 2.13: Sesto prompt

2.8 Sesta versione

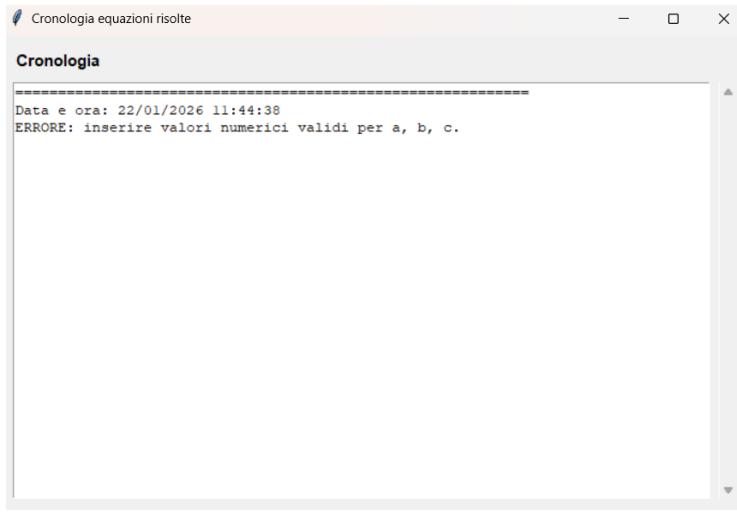


Figura 2.12: Visualizzazione del messaggio di errore nella cronologia

In questa fase del progetto è emersa una criticità funzionale: anche i messaggi di errore (ad esempio input non validi) venivano salvati nel file di cronologia, compromettendo la chiarezza e l'utilità dello storico delle equazioni risolte. Il sesto prompt interviene quindi sulla logica di salvataggio della cronologia, richiedendo che nel file di testo vengano memorizzate esclusivamente le equazioni risolte correttamente, escludendo i casi di errore. L'obiettivo è rendere la cronologia coerente con il suo scopo informativo, limitandola ai soli risultati significativi.

2.9 Settima versione

Il settimo prompt è stato:

Aggiungi la possibilità di visualizzare il grafico della parabola associata all'equazione di secondo grado risolta, mostrando la funzione $f(x) = ax^2 + bx + c$ le eventuali soluzioni reali.

Figura 2.14: Settimo prompt

Il prompt introduce una nuova funzionalità di tipo visuale e analitico, richiedendo l'aggiunta della possibilità di rappresentare graficamente la parabola associata all'equazione di secondo grado risolta. In particolare, viene richiesto di visualizzare la funzione $f(x) = ax^2 + bx + c$ e, se presenti, le soluzioni reali dell'equazione, evidenziandole sul grafico. Questo prompt consente di valutare la capacità dell'AI generativa di integrare strumenti di visualizzazione matematica all'interno di un'applicazione esistente, estendendo il software oltre il solo output numerico. L'aggiunta del grafico migliora l'aspetto didattico e interpretativo dell'applicazione, permettendo all'utente di comprendere in modo immediato il significato geometrico delle soluzioni calcolate.

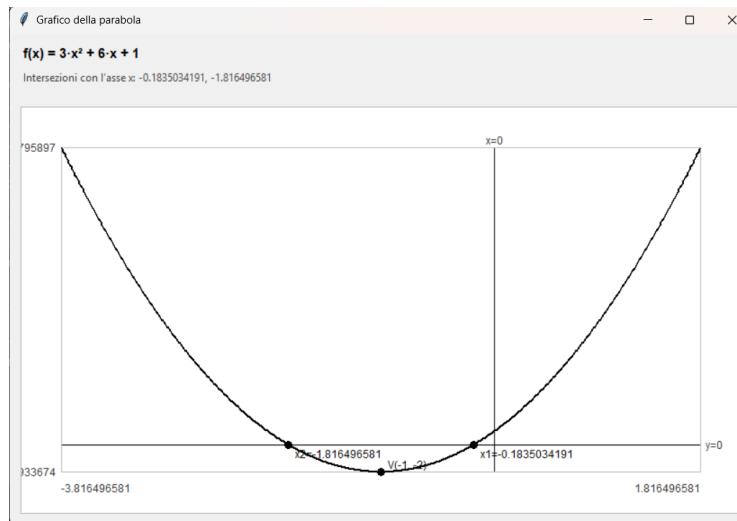


Figura 2.15: Visualizzazione del grafico

Il risultato mostra l'integrazione della visualizzazione grafica della parabola associata all'equazione di secondo grado risolta. L'applicazione apre una finestra dedicata in cui viene rappresentata la funzione $f(x) = ax^2 + bx + c$ consentendo una lettura immediata dell'andamento della curva.

Nel grafico sono evidenziate le intersezioni con l'asse x , corrispondenti alle soluzioni reali dell'equazione, quando presenti, insieme agli assi di riferimento. Questa estensione arricchisce l'applicazione dal punto di vista interpretativo, affiancando all'output numerico una rappresentazione visiva che facilita la comprensione del significato geometrico delle soluzioni.

2.10 Ottava versione

In questa versione è stato dato il seguente prompt:

Migliora l'aspetto grafico dell'applicazione e del grafico della parabola, rendendo l'interfaccia più chiara e gradevole. Utilizza colori più leggibili, font più appropriati per testi, titoli e pulsanti e rendi il layout complessivamente più ordinato.

Figura 2.16: Ottavo prompt

Il prompt richiede un ulteriore intervento di raffinamento dell'interfaccia grafica, esteso sia all'applicazione principale sia alla finestra di visualizzazione del grafico della parabola. L'obiettivo è rendere l'interfaccia più chiara e gradevole dal punto di vista visivo, intervenendo su colori, font, titoli, pulsanti e organizzazione generale del layout.

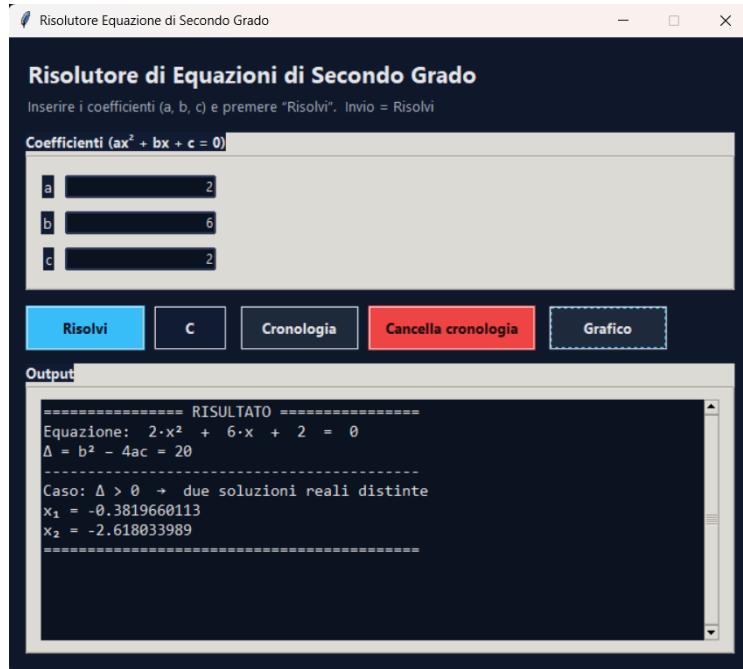


Figura 2.17: Visualizzazione della nuova interfaccia

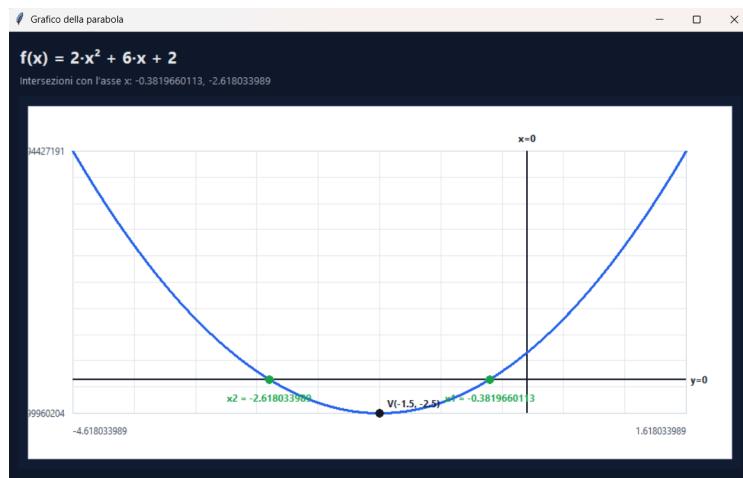


Figura 2.18: Visualizzazione dell’interfaccia nuova per il grafico

Il risultato evidenzia un significativo miglioramento dell’aspetto grafico complessivo dell’applicazione e della finestra di visualizzazione del grafico, senza alterarne le funzionalità. L’interfaccia principale presenta ora una struttura più moderna e coerente, con l’uso di colori più leggibili, font più chiari per titoli, testi e pulsanti, e una migliore organizzazione delle sezioni di input, controllo e output.

Anche il grafico della parabola risulta più chiaro e interpretabile, grazie a una resa visiva più pulita, a una migliore evidenziazione delle soluzioni reali e a una disposizione ordinata degli elementi informativi.

2.11 Nona Versione

Per migliorare ulteriormente l’interfaccia grafica dell’applicazione è stato dato il seguente prompt:

Modifica il tema grafico dell’applicazione rendendolo più chiaro e leggibile, evitando uno stile troppo scuro.
Cambia il font dei testi (titoli, etichette, output e pulsanti) utilizzando un carattere più gradevole e moderno, migliorando la leggibilità complessiva dell’interfaccia.

Figura 2.19: Nono prompt

Il prompt richiede una revisione dell’interfaccia grafica orientata a migliorare la leggibilità complessiva dell’applicazione. In particolare, viene chiesto di abbandonare uno stile cromatico troppo scuro a favore di un tema più chiaro, e di sostituire i font utilizzati con caratteri più moderni e gradevoli, applicandoli in modo coerente a titoli, etichette, pulsanti e area di output.

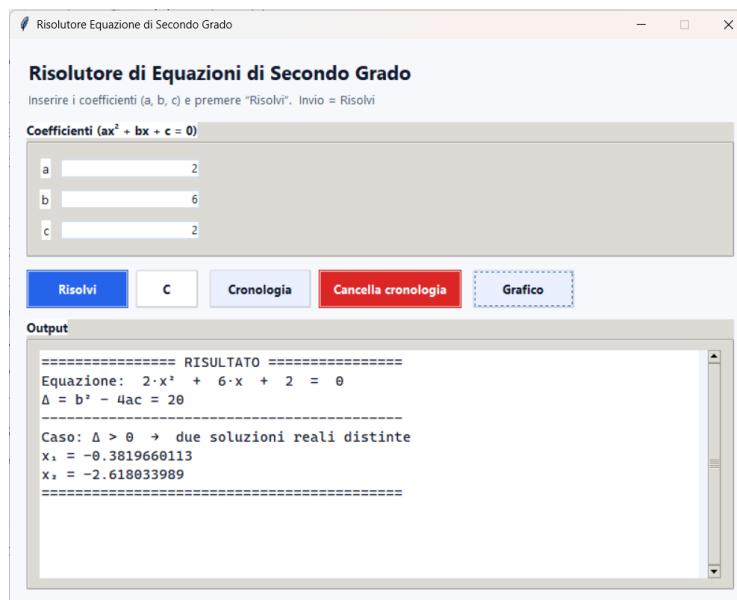


Figura 2.20: Visualizzazione dell'applicazione con la nuova interfaccia

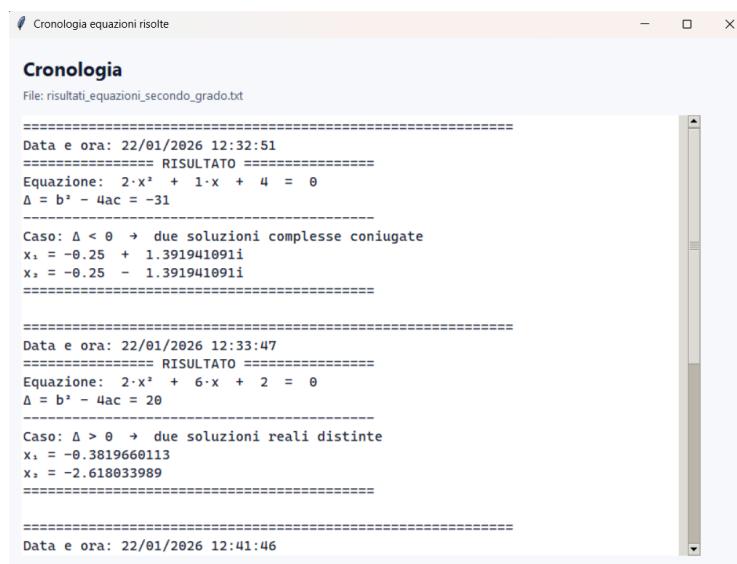


Figura 2.21: Visualizzazione dell'interfaccia nuova per la cronologia

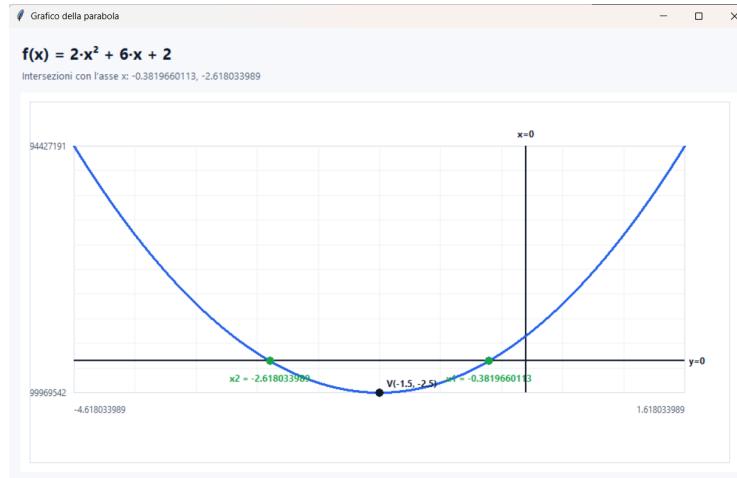


Figura 2.22: Visualizzazione del nuovo grafico

Il risultato mostra l'applicazione di un tema grafico più chiaro e leggibile, in linea con quanto richiesto dal prompt. L'interfaccia principale presenta ora colori più neutri e luminosi, che migliorano la leggibilità dei contenuti e riducono l'affaticamento visivo rispetto alle versioni precedenti. Anche la scelta dei font risulta più moderna e uniforme, applicata in modo coerente a titoli, etichette, pulsanti e area di output, contribuendo a una migliore gerarchia visiva delle informazioni. La finestra del grafico mantiene la chiarezza delle informazioni matematiche, con una resa più pulita e ordinata degli elementi grafici.

2.12 Decima e ultima versione

Aggiungi la possibilità di visualizzare i passaggi di calcolo per la risoluzione dell'equazione di secondo grado tramite un altro bottone. Il procedimento deve mostrare il calcolo del discriminante e l'applicazione della formula risolutiva.

Figura 2.23: Prompt finale

L'ultimo prompt richiede l'aggiunta di una funzionalità a carattere esplicativo e didattico, introducendo un pulsante dedicato alla visualizzazione dei passag-

gi di calcolo per la risoluzione dell'equazione di secondo grado. In particolare, viene richiesto di mostrare in modo esplicito il calcolo del discriminante e l'applicazione della formula risolutiva, illustrando il procedimento seguito dall'applicazione.

Il prompt consente di valutare la capacità dell'AI generativa di integrare una forma di spiegazione strutturata del risultato, affiancando all'output finale una descrizione del processo matematico sottostante.

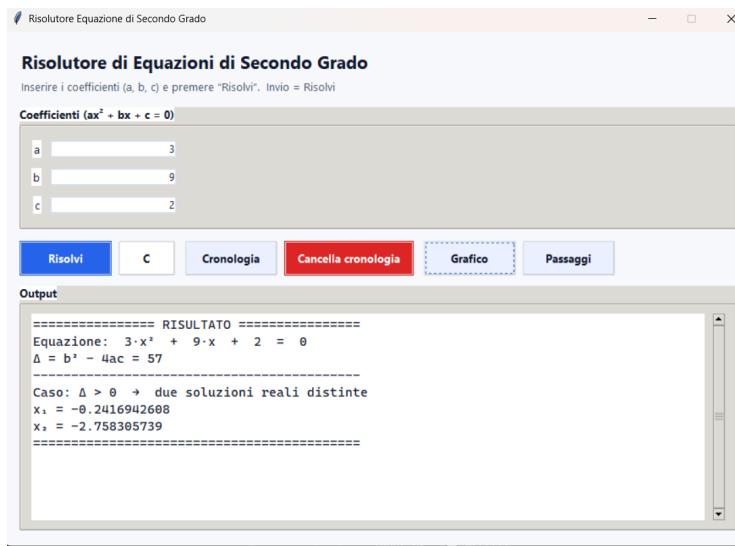


Figura 2.24: Visualizzazione del tasto del procedimento

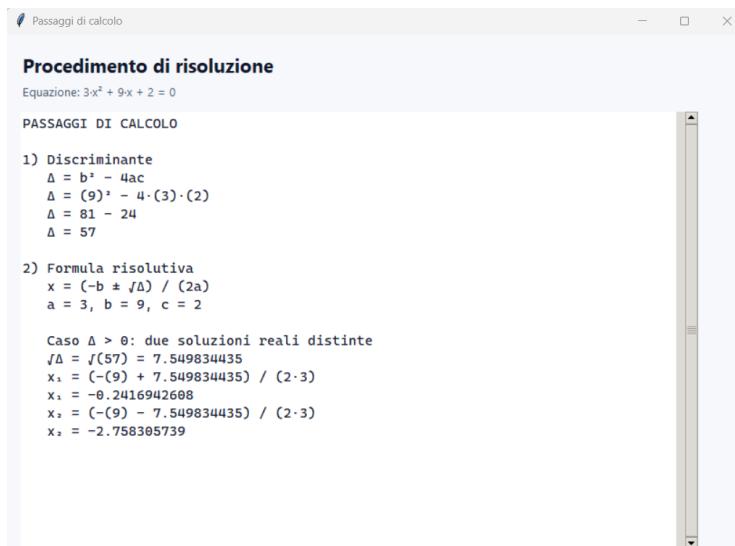


Figura 2.25: Visualizzazione del procedimento dell'equazione

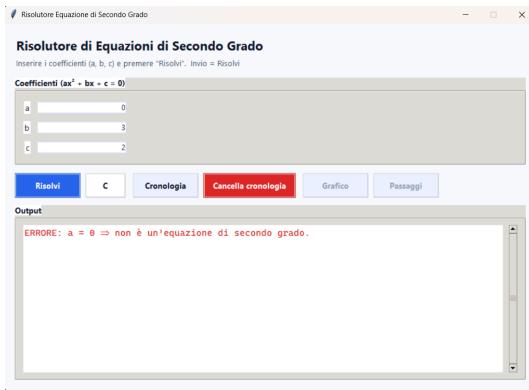


Figura 2.26: Visualizzazione della gestione del caso $a = 0$

Il risultato mostra l'aggiunta di una finestra dedicata alla visualizzazione dei passaggi di calcolo, accessibile tramite un nuovo pulsante nell'interfaccia principale. L'applicazione presenta in modo strutturato il procedimento di risoluzione dell'equazione, esplicitando il calcolo del discriminante e l'applicazione della formula risolutiva, con i valori intermedi utilizzati.

La rappresentazione testuale del procedimento è chiara e ordinata, coerente con i dati inseriti e con le soluzioni finali mostrate nell'output principale.

L'immagine finale mostra invece la corretta gestione del caso particolare $a = 0$, per il quale l'equazione inserita non è più di secondo grado. In questa condizione, l'applicazione intercetta l'input prima di eseguire il calcolo e restituisce un messaggio di errore chiaro ed esplicito nell'area di output, segnalando all'utente che non si tratta di un'equazione di secondo grado.

2.13 Conclusioni

Il progetto ha dimostrato come l'AI generativa possa essere efficacemente impiegata come strumento di supporto nello sviluppo di applicazioni software, all'interno di un contesto di Data Science. Attraverso una sequenza progressiva di prompt, è stato possibile progettare, estendere e migliorare un risolutore di equazioni di secondo grado, intervenendo sia sugli aspetti funzionali sia su quelli di usabilità e presentazione.

L’AI generativa si è rivelata utile nella traduzione di requisiti espressi in linguaggio naturale in codice eseguibile, nella gestione di casi limite, nell’integrazione di funzionalità avanzate (cronologia, grafici, passaggi di calcolo) e nel miglioramento dell’interfaccia grafica, mantenendo il controllo umano sulle scelte progettuali e sulla validazione dei risultati.

Nel complesso, l’esperienza conferma che l’AI generativa, se utilizzata in modo consapevole e guidato, rappresenta un valido supporto allo sviluppo software e alla comunicazione dei risultati, affiancando il data scientist e lo sviluppatore senza sostituirne il ruolo decisionale.

Capitolo 3

Attività di prompt engineering per la generazione di una dashboard

3.1 Modello utilizzato

Per lo sviluppo di questo secondo task, ovvero la generazione di una dashboard tramite l'uso di AI generativa, è stato utilizzato il modello **Nano Banana** di **Gemini**, Google, che rappresenta uno dei più noti modelli attualmente in circolazione per la generazione di immagini complesse e strutturate basate sull'intelligenza artificiale.

Il lavoro è stato sviluppato seguendo un approccio iterativo di prompt engineering. In una prima fase sono stati forniti al modello lo schema E-R e il contesto applicativo. Successivamente, i prompt sono stati progressivamente raffinati per migliorare la coerenza delle dashboard rispetto al modello dati, la chiarezza visiva e la significatività delle informazioni rappresentate. Dopo ogni iterazione sono stati analizzati i risultati ottenuti, individuandone punti di forza e limiti, e adattando di conseguenza i prompt successivi.

3.2 Diagramma E-R di partenza



Figura 3.1: Schema logico di partenza per la dashboard

Lo schema logico presentato delinea un’architettura relazionale per la gestione semplificata di una biblioteca universitaria, strutturata su tre entità principali che interagiscono attraverso vincoli di integrità referenziale. La tabella **Utenti** e la tabella **Libri** fungono da anagrafiche di base, identificate dalle relative chiavi primarie, le quali garantiscono l’univocità di ogni record all’interno del sistema. Il fulcro operativo della struttura è rappresentato dalla tabella **Prenotazioni**, che agisce come entità di collegamento e memorizza i dettagli transazionali come la data di prenotazione, la scadenza e lo stato attuale dell’operazione.

La coerenza dei dati è assicurata dall’implementazione delle chiavi esterne (FK) all’interno della tabella **Prenotazioni**: il campo **ID_Utente** collega ogni transazione a un profilo specifico, realizzando la relazione uno a molti tra chi richiede il servizio e le richieste effettuate. Parallelamente, il campo **ID_Libro** associa la prenotazione a un volume del catalogo, permettendo al sistema di aggiornare e monitorare l’attributo di disponibilità del testo.

3.3 Dashboard per la tabella Utenti

3.3.1 Primo prompt

Primo prompt _____

Supponiamo di avere una base di dati per la gestione di una
biblioteca universitaria, organizzata secondo il

seguente schema logico:

UTENTE: ID_Utente (PK), Nome, Cognome, Email, Tipo

LIBRO: ID_Libro (PK), Titolo, Autore, Genere,
AnnoPubblicazione, Disponibile

PRENOTAZIONE: ID_Prenotazione (PK),
DataPrenotazione, DataScadenza, Stato

Un Utente può effettuare molte Prenotazioni

Ogni Prenotazione è associata a un solo Utente

Un Libro può comparire in molte Prenotazioni

Ogni Prenotazione fa riferimento a un solo Libro

Inoltre, per "Tipo" in "UTENTE" si intende
la tipologia di utente ovvero studente,
docente, esterno, ecc.

D'ora in avanti ti chiederò di generare
immagini di dashboard, facendo sempre
riferimento a questo schema logico.

Iniziamo con l'analisi dell'utenza
quindi genera un'immagine di una dashboard
per l'entità UTENTE di una biblioteca
universitaria.

La dashboard deve mostrare:

- Numero totale di utenti

- Numero di utenti attivi
(che hanno almeno una prenotazione)
- Distribuzione degli utenti per tipo
- Un grafico che mostri il numero di prenotazioni per utente

Stile professionale, layout ordinato, moderno e pulito, come se fosse progettata da un data analyst professionista.

Ovviamente tutto quello che comparirà nella dashboard deve essere coerente con i dati del modello logico che ti ho descritto.

Infine, se ritieni opportuno che la dashboard contenga anche altri grafici oltre a quelli che ti ho proposto sei libero di aggiungerli o migliorare opportunamente le mie proposte



Figura 3.2: Risultato primo prompt

Il risultato è subito buono ma presenta sicuramente diverse cose da migliorare come ad esempio:

- Si potrebbe aumentare la variabilità di colori
- I nomi sono confusi e poco leggibili, laddove si riescono a vedere si nota che non si tratta comunque di nomi di utenti
- La dashboard è stata generata all'interno di un monitor, cosa da evitare
- Pur avendo seguito il consiglio nel prompt di migliorare la proposta con nuovi grafici, il grafico inserito in basso a destra non fornisce nuove informazioni rilevanti rispetto a quelle già presenti

Si passa dunque al secondo prompt

3.3.2 Secondo prompt

Secondo prompt

Va bene ma cerca di inserire più colori
nei grafici in modo che siano più chiari
e piacevoli da vedere, non solo il blu.
Inoltre non si leggono i nomi fintizi
degli utenti, vorrei che si leggessero bene.
Altro appunto, al posto del grafico
"Utenti più attivi", dato che
l'informazione che se ne ricava
è molto simile al grafico della
top 20 per prenotazione, inserisci
al suo posto un grafico che traccia
l'andamento delle nuove registrazioni
degli utenti in biblioteca.
Infine voglio che la dashboard occupi

tutta l'immagine e non stia all'interno
di un monitor come hai fatto.

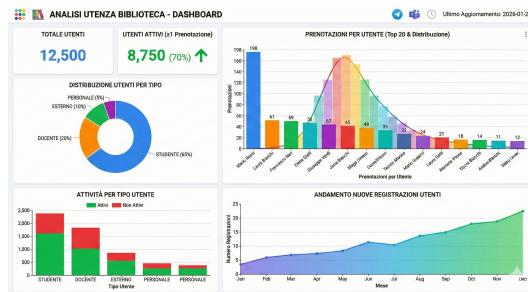


Figura 3.3: Risultato secondo prompt

Il risultato è già notevolmente migliorato, rimangono come uniche problematiche la doppia colonna "Personale" nel grafico in basso a sinistra, i nomi ancora illegibili e l'aggiunta delle icone di "Teams" e "Telegram" sulla barra in alto che ruchiamano ancora l'aspetto di un computer.

3.3.3 Terzo prompt

Terzo prompt

Va bene, ma nel grafico
"Attività per utente" ci sono
due colonne per "Personale",
mettine solo una.

Alcuni nomi del grafico
"Prenotazioni per utente" non si
leggono ancora, aumenta la qualità
dell'immagine o comunque
definiscili bene.

Rimuovi le icone di teams
e telegram dalla barra in alto

e lascia solo il titolo,
sistemandolo meglio

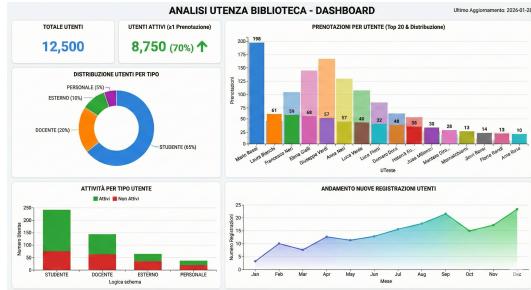


Figura 3.4: Risultato terzo prompt

In questo caso emerge un problema di coerenza con i dati piuttosto che grafico. Infatti la scala del grafico in basso a sinistra è sproporzionata rispetto al numero totale e al numero di utenti attivi che la dashboard indica. Nel prossimo prompt si cerca di migliorare quest'aspetto.

3.3.4 Quarto prompt

Quarto prompt

Va bene, ultime modifiche:

Rimetti la curva della

distribuzione in "Prenotazioni per utente"

In "attività per tipo utente"

i dati della scala del grafico

(max 250) non rispecchiano il

totale degli utenti (12500).

Cerca di essere più coerente.

Il resto può rimanere esattamente così.

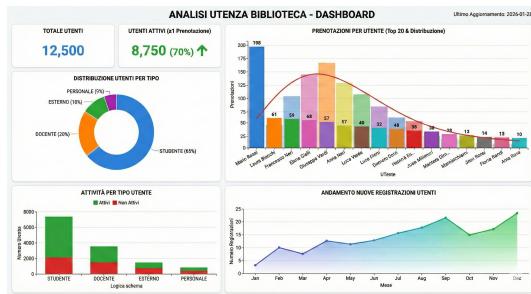


Figura 3.5: Risultato quarto prompt

A questo punto ci si accorge che nel grafico in alto a destra, seppur risultano in ordine come altezza le barre, non coincide il numero che indica la quantità (si passa da 59 a 68, rompendo l'ordine).

3.3.5 Quinto prompt

Quinto prompt

Ultima modifica: i numeri delle "prenotazioni per utente" devono andare dal più grande al più piccolo nella top 20, tu invece hai messo 61 e 59 prima di 69, correggi. Sistema meglio anche la parte della distribuzione.
Tutto il resto deve rimanere com'è

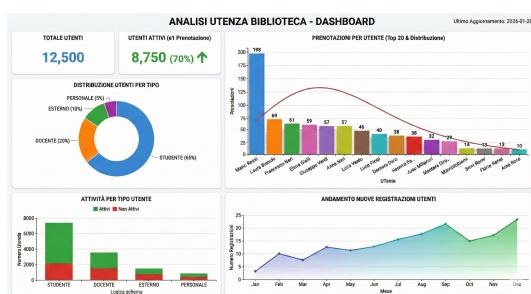


Figura 3.6: Risultato quinto prompt

Il grafico è stato sistemato ma risultano ora confuse alcune etichette dei vari grafici che verrano richieste nel prompt successivo.

3.3.6 Sesto prompt

Sesto prompt

Correggi e rendi leggibili
le etichette:
"Numero utenza" per l'asse y
di "attività per tipo utente",
"Prenotazioni" per l'asse y di
"Prenotazioni per utente" ,
"Numero registrazioni" per
l'asse y di "Andamento nuove
registrazioni utenti".
Correggile e rendile leggibili.
Anche i nomi degli utenti
non si leggono ancora, scrivili bene.
Infine correggi anche l'etichetta
"Logica schema"che non ha senso,
mettici "Tipo utenti" al suo posto
e "UTenti" sulla x di
"Prenotazioni per utente",
scrivilo correttamente "Utenti"

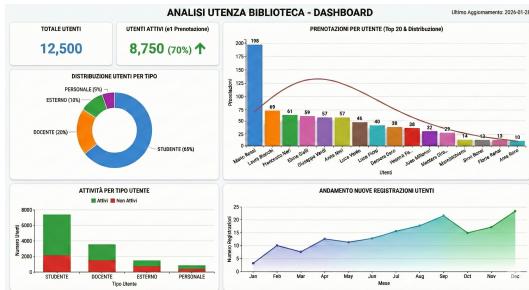


Figura 3.7: Risultato sesto prompt

Questo chiude la prima dashboard. Nel complesso il risultato è buono e coerente, tramite i vari prompt si è riusciti a correggere quasi tutti gli errori e i problemi. Rimane piuttosto evidente il problema della leggibilità dei nomi fintizi degli utenti, limite che sembra il modello non riuscire a superare.

3.4 Dashboard per la tabella Libri

3.4.1 Primo prompt

Primo prompt

Passiamo ora all'analisi del catalogo dei libri. Genera quindi un'immagine di una dashboard per l'entità LIBRO di una biblioteca universitaria.

La dashboard deve mostrare:

- Numero totale di libri
- Numero di libri disponibili vs non disponibili
- Top 10 libri più prenotati
- Distribuzione dei libri per genere

Anche in questo caso usa grafici

chiari e una struttura da dashboard aziendale, tenendo sempre a mente che tutto deve essere ricavabile dal modello E-R che ti ho fornito all'inizio.

Infine, se necessiti di voler cambiare in meglio o aggiungere qualche altro grafico rispetto a quelli descritti, fai pure.

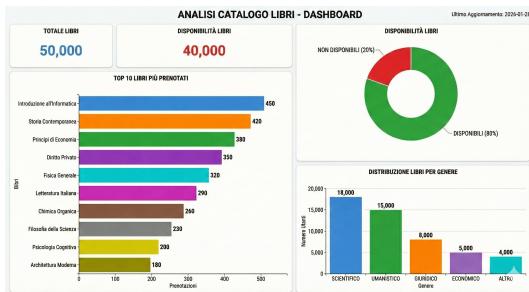


Figura 3.8: Risultato Primo prompt

Avendo già generato la dashboard precedente, il risultato è subito migliore e vicino a quello ottimale. Anche in questo caso il problema principale è legato alle etichette incorrecte e poco leggibili.

3.4.2 Secondo prompt

Secondo prompt

Sistema l'etichetta "Libri" in
 "Top 10 libri più prenotati" e al
 posto di "Numero utenti" va
 "Numero libri" in "Distribuzione
 libri per genere".

Inserisci anche un grafico che faccia il confronto tra i generi più presenti in biblioteca e i generi più prenotati

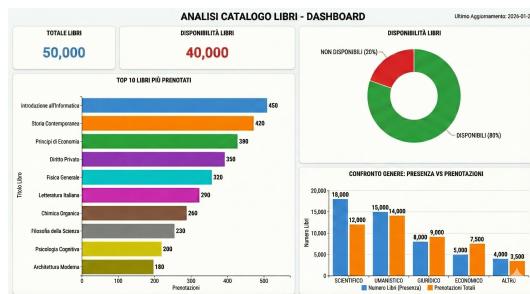


Figura 3.9: Risultato secondo prompt

A questo punto il risultato può già considerarsi ottimale, è stato migliorato anche il grafico in basso a destra e reso più informativo.

3.5 Dashboard per la tabella Prenotazioni

3.5.1 Primo prompt

Primo prompt

Passiamo ora all'analisi delle prenotazioni. Genera un'immagine di una dashboard per l'entità PRENOTAZIONE di una biblioteca universitaria.

La dashboard deve mostrare:

- Numero totale di prenotazioni
- Prenotazioni attive,
- scadute e restituite

- Andamento delle prenotazioni nel tempo
- Tabella riepilogativa delle prenotazioni

Stile e numeri coerenti con le altre dashboard e professionale.
Anche in questo caso ricorda il modello E-R e sentiti libero di aggiungere/modificare in modo adeguato le mie proposte

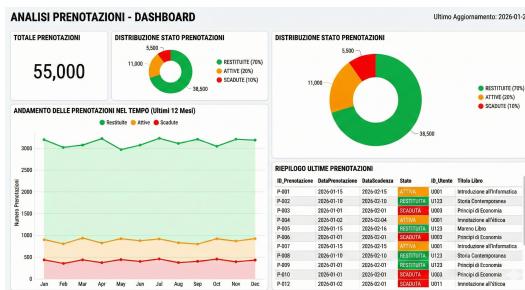


Figura 3.10: Risultato primo prompt

Anche in questo caso il risultato è già buono ma un problema evidente è sicuramente la duplicazione del grafico "Distribuzione stato prenotazioni". Inoltre potrebbe essere più verosimile accentuare i picchi delle richieste di prenotazioni nei mesi di sessione d'esame.

3.5.2 Secondo prompt

Secondo prompt

Hai fatto due grafici identici per "Distribuzione stato prenotazione", ne basta uno. Sostituisci l'altro con

un grafico che mostri la durata media delle prenotazioni.

Inoltre migliora il grafico dell'andamento delle prenotazioni nel tempo cercando di renderlo più variabile, accentuando maggiormente i picchi intorno ai mesi di sessione di esame

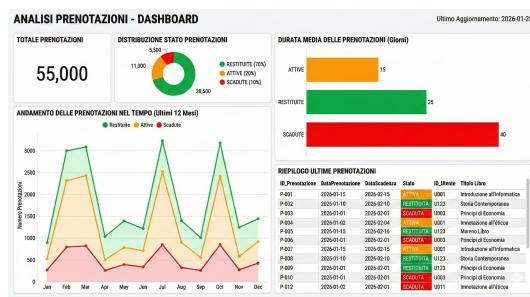


Figura 3.11: Risultato secondo prompt

Come si vede, i picchi sono comparsi ma non perfettamente in corrispondenza dei mesi di sessione e il nuovo grafico dovrebbe riguardare solo le prenotazioni restituite.

3.5.3 Terzo prompt

Terzo prompt

I mesi di sessione sono Gennaio,
Febbraio, Giugno, Luglio e Settembre.
Il grafico della durata media
rendilo un po' più carino e inoltre
si deve riferire solo a quelle restituite.

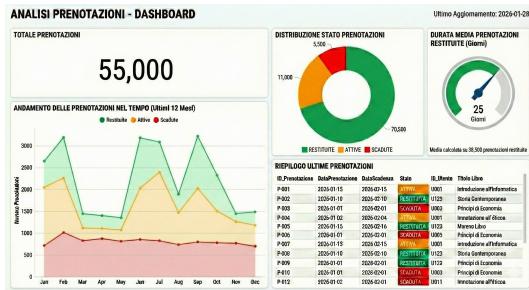


Figura 3.12: Risultato terzo prompt

A questo punto rimane solo da sistemare la tabella che è stata inspiegabilmente resa poco leggibile e sfocata nelle colonne di stato.

3.5.4 Quarto prompt

Quarto prompt

Perfetto, ultima modifica:

rendi più leggibile la tabella,
la cella dello "stato" è molto sfocata

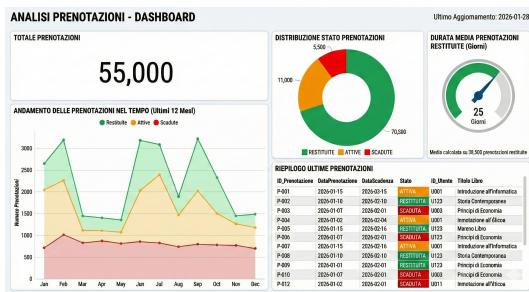


Figura 3.13: Risultato quarto prompt

Si è così raggiunto il risultato ottimale, la tabella è ora leggibile e chiara, così come anche le etichette e i grafici.

3.6 Conclusioni

I risultati ottenuti mostrano che, attraverso un processo iterativo di prompt engineering, è possibile guidare il modello generativo verso la produzione di dashboard complessivamente corrette e coerenti con il contesto applicativo. Dopo alcuni raffinamenti, il modello riesce a rappresentare in modo efficace le informazioni principali. Tuttavia, emergono alcune criticità ricorrenti, in particolare nella gestione delle etichette, nella denominazione degli elementi e nella coerenza di alcuni dettagli informativi. Soprattutto la gestione del testo è problematica e spesso i nomi sono sbagliati e poco leggibili e in diversi casi è stato necessario intervenire sui prompt per correggere imprecisioni o ambiguità, evidenziando come la qualità del risultato dipenda fortemente dalla precisione delle istruzioni fornite. Talvolta però, nemmeno la precisione delle istruzioni è sufficiente a superare tali limiti, confermando che, almeno in questo campo, i modelli generativi rappresentano un valido supporto, ma non sostituiscono la progettazione e il controllo umano.