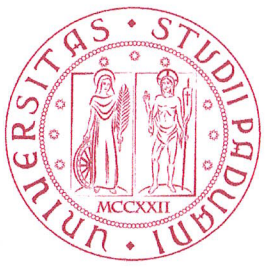
Verbale Riunione 2024-04-09

Gruppo Argo — ***Progetto ChatSQL***

Informazioni sul documento	
Versione	1.0.0
Approvazione	Riccardo Cavalli
	Zucchetti S.p.A.
Uso	Esterno
Distribuzione	Prof. Tullio Vardanega
	Prof. Riccardo Cardin
	Gruppo Argo



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Registro delle modifiche

Ver.	Data	Redazione	Verifica	Descrizione
1.0.0	2024-04-30	Riccardo Cavalli	Riccardo Cavalli	Verifica generale e approvazione del documento
0.0.2	2024-04-25	Sebastiano Lewental	Raul Pianon	Modifiche generali dopo l'aggiornamento del template *LaTeX*$_G$
0.0.1	2024-04-18	Sebastiano Lewental	Raul Pianon	Stesura del documento

Indice

1 Informazioni

- **Inizio incontro:** 10:30
- **Fine incontro:** 11:40
- **Pianificazione incontro:** Mail
- **Tipo incontro:** remoto (Zoom)

1.1 Descrizione

Questo verbale riporta varie domande poste alla *Proponente*$_G$ a proposito di analisi e progettazione del *capitolato*$_G$ con lo scopo di chiarire i *requisiti*$_G$ del progetto.

1.2 Partecipanti

- **Argo:**
  - Tommaso Stocco
  - Marco Cristo
  - Raul Pianon
  - Sebastiano Lewental
  - Martina Dall'Amico
  - Riccardo Cavalli
  - Mattia Zecchinato
- **Zucchetti S.p.A.:**
  - Gregorio Piccoli

1.3 Glossario

Allo scopo di evitare incomprensioni relative al linguaggio utilizzato nella documentazione di progetto, viene fornito un *Glossario*, nel quale ciascun termine è corredato da una spiegazione che mira a disambiguare il suo significato. I termini tecnici, gli acronimi e i vocaboli ritenuti ambigui vengono formattati in corsivo all'interno dei rispettivi documenti e marcati con una lettera $_G$ in pedice. In questo documento viene formattata solamente la prima ricorrenza di un termine definito nel *Glossario*.

2 Riunione

2.1 Ritrovo iniziale

Il gruppo si è riunito circa mezz'ora prima dell'appuntamento con l'azienda per discutere la chiarezza delle domande e organizzare una scaletta. Sono stati formulati inoltre dei quesiti aggiuntivi per garantire un'interazione più fluida con la Proponente.

Il gruppo ha affidato il compito di moderare l'incontro al responsabile in carica, delegando la discussione sui *casi d'uso*$_G$ agli analisti. Come concordato per via telematica, il referente di Zucchetti si è unito al meeting alle ore 10:30.

2.2 Argomenti e temi dell'incontro

2.2.1 Diagramma dei casi d'uso

Domanda: È stata ideata una bozza riguardante i *casi d'uso*$_G$ per avere un riscontro dalla Proponente e capire se rispetta la visione dell'applicazione proposta nel capitolato. Nel documento sono stati pensati tre *attori*$_G$ principali: un Utente Generico, un Utente Non Autenticato e un Tecnico. L'Utente Generico può visualizzare il *prompt*$_G$ finale a seguito dell'inserimento della richiesta in linguaggio naturale e della selezione del *dizionario dati*$_G$. La visualizzazione della lista dei *dizionari dati*$_G$ è sempre possibile per l'Utente Generico. Quest'ultimo, autenticandosi, acquisisce permessi aggiuntivi e diventa un Tecnico. La motivazione di questa scelta deriva dalla seguente considerazione: un Utente Generico, una volta autenticato, diviene un Tecnico, il quale non ha più opportunità di fare il login, ma solo il logout. Se un Tecnico ereditasse le funzionalità direttamente da un Utente Generico, dotato dell'opportunità di autenticarsi, questa sarebbe disponibile anche al Tecnico. Da qui la necessità di distinzione tra Utente Generico e Utente Non Autenticato.

Risposta: La Proponente suggerisce di rimanere attinenti ai metodi appresi nel corso di Ingegneria del Software, mantenendo come priorità il rispetto dei requisiti minimi del capitolato.

2.2.2 Gestione login per il Tecnico

Domanda: La Proponente suggerisce un metodo diverso da quello proposto per separare la funzione di login dalle funzioni proprie del Tecnico?

Risposta: Nello schema proposto dai fornitori, è ragionevole che l'Utente Generico non sia interessato al login, tuttavia la soluzione ottimale sarebbe poterlo fare per ogni *attore*$_G$. Quindi, accedendo alla *web app*$_G$, ci si dovrebbe poter autenticare a priori. Ciò che è stato ideato dai fornitori nei casi d'uso ammette l'interazione da parte dell'Utente Generico con l'applicazione senza autenticazione. La Proponente sarebbe favorevole ad estendere la funzione di login anche per questo attore, tenendo come riferimento prioritario le nozioni fornite dal corso.

2.2.3 Login per l'Utente Generico

Domanda: C'è la preferenza di autenticare anche l'Utente Generico?

Risposta: La Proponente non ha preferenze specifiche, sottolinea però che se l'utente dovesse consumare risorse costose, come delle chiamate *API*$_G$ a pagamento, sarebbe preferibile limitare l'accesso tramite il login. Perciò, se i fornitori decidessero di implementare il *requisito*$_G$ opzionale delle chiamate API (sottoscrivendo un abbonamento a pagamento), allora la Proponente consiglierebbe di autenticare anche l'Utente Generico. Per il Tecnico invece, siccome i dati verranno resi in qualche modo *persistenti*$_G$ e richiedono spazio di archiviazione, la funzione di login è necessaria.

2.2.4 Numero di attori

Domanda: La necessità della Proponente è quindi di avere solamente due attori?

Risposta: Sì, due attori sono adeguati.

2.2.5 Ulteriori funzionalità per l'utente Tecnico

Domanda: Il Tecnico necessita di ulteriori funzionalità?

Risposta: Alcuni gruppi del primo lotto hanno stabilito che il Tecnico può svolgere anche le attività proprie dell'Utente Generico, ma, quando invia delle richieste, potrebbe aver bisogno di una funzionalità di *debug*$_G$. La struttura del dizionario dati sarà a discrezione dei fornitori e conterrà la lista delle tabelle e le relazioni tra di esse. Inoltre sarà presente sia una descrizione tecnica che una descrizione in linguaggio naturale: esse entreranno in relazione tra loro quando elaborate dal *modello*$_G$. In seguito alla definizione del dizionario dati, il Tecnico potrebbe aver bisogno di eseguire dei test per avere un riscontro sul legame tra queste descrizioni e la richiesta. Di conseguenza, una funzionalità di debug sarebbe utile al Tecnico per capire come il dizionario dati si armonizza con il modello per la generazione del prompt.

2.2.6 Pre-prompt

Domanda: Quanto è importante il *prompt engineering*$_G$?

Risposta: Quando il dizionario dati è di dimensioni ridotte (ad esempio un *dizionario*$_G$ con tre tabelle), quest'ultimo può essere contenuto interamente all'interno del prompt. Tuttavia, se il dizionario è di grandi dimensioni, si presentano dei problemi, tra cui il costo elevato dell'elaborazione di un prompt con numerosi *token*$_G$. A livello teorico è possibile avere prompt di grandi dimensioni ma è meglio mantenerli asciutti, stabilendo quali siano le sezioni pertinenti alla richiesta dell'utente. Diventa così necessario capire il metodo con cui estrarre le porzioni rilevanti dal dizionario dati durante la creazione del prompt. A dimostrazione di ciò, chiedendo a un *LLM*$_G$ di elaborare un prompt e aggiungendo progressivamente dettagli superflui, è evidente come il modello generi dati incoerenti.

2.2.7 Funzionalità di debug

Domanda: Il debug è uno stato intermedio rispetto a quanto visto?

Risposta: Sì, il debug è utile a capire come è stato generato il prompt e perché sono presenti determinate porzioni del dizionario dati.

2.2.8 Stima della correttezza del prompt

Domanda: Come si può stimare la correttezza del prompt?

Risposta: La richiesta dell'utente viene inserita nella sezione finale del prompt. Quello che ci interessa è il segmento precedente, ovvero la porzione del dizionario dati attinente alla richiesta in linguaggio naturale. Bisogna capire come si è arrivati all'estrazione di quella porzione che viene prefissata alla richiesta.

2.2.9 Visualizzazione debug

Domanda: Per il debug, si può sovrascrivere la funzionalità di visualizzazione del prompt quando l'attore che lo richiede è un Tecnico. È l'approccio corretto?

Risposta: Un'opzione sarebbe estendere il *caso d'uso*$_G$ di partenza, per gestire la situazione in cui il Tecnico è interessato a ottenere, oltre al prompt, un *report*$_G$ sull'interazione tra il modello e il dizionario dati.

2.2.10 Modalità di visualizzazione per l'utente Tecnico

Domanda: Il Tecnico possiede solo la visualizzazione con *debugging*$_G$ o anche quella normale?

Risposta: Entrambe, perché sono due circostanze differenti. Se il dizionario è molto piccolo, la soluzione è semplice. Quando invece il dizionario è grande, il prompt può diventare fuorviante per il modello se contiene (o non contiene) determinate tabelle. Facendo una controprova ed eseguendo delle richieste in cui i dati forniti sono differenti dagli stretti necessari, si osserva che quando i dati sono in eccesso, il modello si confonde, quando invece mancano, genera dati casuali. La funzionalità di debug può aiutare il Tecnico a capire come riformulare il dizionario dati affinché l'applicazione restituisca il prompt desiderato.

2.2.11 Utilizzo di txtai in locale

Domanda: Durante il primo incontro, la Proponente aveva consigliato *txtai*$_G$. È collegato alla fase di *ricerca semantica*$_G$?

Risposta: Sì. Quello che è interessante è che la ricerca semantica in txtai opera tramite la *sentence similarity*$_G$; inoltre, si possono usare *modelli*$_G$ molto piccoli ad uso di macchine meno potenti. Utilizzando un modello specifico, che viene fatto girare su un computer della Proponente attrezzato con una GPU RTX 3060, si può osservare che, nonostante venga eseguito in locale, non ci sono ostacoli al funzionamento o problemi di surriscaldamento.

2.2.12 LLM di Hugging Face

Domanda: Da dove è stato selezionato il modello utilizzato nell'esempio precedente?

Risposta: Da *Hugging Face*$_G$. Si tratta della versione quantizzata di *OpenChat*$_G$ in formato *GGUF*$_G$.

2.2.13 Implementazione lingue alternative

Domanda: Per adottare lingue diverse da quella del dizionario dati, basterebbero dei sinonimi nella lingua alternativa, inseriti all'interno del dizionario?

Risposta: A tal proposito si osserva che, eseguendo una richiesta in lingua inglese e fornendo una descrizione delle tabelle in italiano, il modello opera con successo. Il modello è in grado di dedurre da sé cosa è richiesto, però bisogna ragionare su come trattare il dizionario dati avendo una richiesta in una lingua differente.

2.2.14 Gestione delle richieste in lingue alternative

Domanda: Avendo un dizionario dati da filtrare in base alla richiesta, quando arriva una richiesta in russo, è necessario gestire il filtraggio con la lingua corrispondente?

Risposta: Si possono caricare *dizionari*$_G$ in lingue differenti, altrimenti, utilizzando tx-tai, non ce n'è bisogno. In alternativa, potreste impiegare dei modelli di traduzione, che sono solitamente molto piccoli. Su Hugging Face, per esempio, esistono i modelli di traduzione Helsinki. Impiegando il modello Mistral, è possibile tradurre una richiesta dalla lingua italiana a quella inglese, fornendo la *query*$_G$ *SQL*$_G$ corretta.

2.2.15 Valutazione criticità lingue alternative

Domanda: A questo punto l'applicazione non diventa poco estendibile, perché è necessario ragionare prima per valutare le lingue disponibili?

Risposta: Sì. Altrimenti è possibile tradurre direttamente la richiesta, ma non è un requisito obbligatorio.

2.2.16 Metodologia Agile

Domanda: I fornitori valutano l'adozione di una metodologia *Agile*$_G$ per la gestione di progetto. Sarebbe quindi prevista una riunione alla fine di ogni *sprint*$_G$ (della durata di circa 2 settimane), sia interna che con il cliente. Secondo la Proponente è fattibile organizzare un incontro ogni due settimane?

Risposta: Sì. La Proponente attende una richiesta tramite email per programmare gli incontri. Tuttavia, la Proponente consiglia al gruppo di non impiegare una metodologia troppo stringente, specialmente per quanto riguarda l'organizzazione delle riunioni.

2.2.17 Feedback

Domanda: Ci sono delle attività su cui porre l'attenzione nel prossimo sprint?

Risposta: La sezione riguardante i casi d'uso è stata adeguatamente approfondita; sarà utile concentrarsi sull'apprendimento del processo di generazione del prompt. Inoltre, è essenziale sviscerare l'interazione tra il dizionario dati, la richiesta dell'utente e la generazione del prompt.

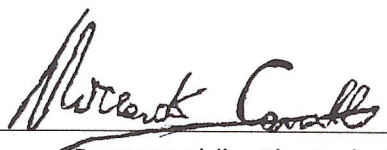
3 Todo / In Progress

Issue	Incarico	Incaricato/a	Scadenza
#27	Stesura verbale riunione	Sebastiano Lewental	2024-04-18
-	Revisione della separazione e relazione tra gli attori principali del sistema	Marco Cristo, Martina Dall'Amico, Sebastiano Lewental	2024-04-16
#38	Espansione del caso d'uso relativo alla funzionalità di debug	Marco Cristo, Martina Dall'Amico, Sebastiano Lewental, Mattia Zecchinato	2024-04-16
#39	Analisi del modello di sviluppo corrente e di eventuali alternative	Riccardo Cavalli	2024-04-16

3.1 Prossima riunione

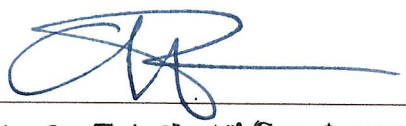
La prossima riunione esterna è fissata per la fine del secondo sprint, affinché il gruppo possa disporre di una finestra temporale più ampia per ragionare sui temi discussi durante l'incontro.

Luogo e Data:
Padova (PD) 2024-04-09

Firma: 

Responsabile: Riccardo Cavalli

Per approvazione:

Firma: 

Referente: Greg[illegible] (Zucchetti S.p.A.)