

Valutazione Capitolati

Contatti:

swateng.team@gmail.com

Versione 1.5



Descrizione

In questo documento si valutano gli aspetti positivi e negativi di ciascun capitolato. I capitolati in esame sono riportati al seguente link:

<https://www.math.unipd.it/~tullio/IS-1/2023/Progetto/Capitolati.html>

Registro delle Modifiche

Versione	Data	Descrizione	Autore	Ruolo
1.5	25-10-2023	Revisione completa	Simone Caregnato	Revisore
1.4	24-10-2023	Revisione completa	Giacomo D'Ovidio	Revisore
1.3	23-10-2023	Stesura capitolati C4, C9 e rifinito i rimanenti	Riccardo Toniolo	Redattore
1.2	22-10-2023	Stesura capitolati C6, C7 e alcuni punti dei rimanenti	Matteo Rango	Redattore
1.1	21-10-2023	Stesura capitolati C2, C5 e alcuni punti dei rimanenti	Riccardo Costantin	Redattore
1.0	20-10-2023	Definita struttura del file, stesura capitolati C1, C8	Nancy Kalaj	Redattore

Indice

1 Valutazione del capitolato scelto.....	4
1.1 Capitolato C6 - InnovaCity: Smart city monitoring platform.....	4
2 Valutazione dei capitolati rimanenti.....	7
2.1 Capitolato C1 - Knowledge management AI.....	7
2.2 Capitolato C2 - Sistemi di raccomandazione.....	8
2.3 Capitolato C3 - Easy meal.....	9
2.4 Capitolato C4 - A ChatGPT plugin with Nuvolaris.....	10
2.5 Capitolato C5 - WMS3: warehouse management 3D.....	11
2.6 Capitolato C7 - ChatGPT vs BedRock developer analysis.....	13
2.7 Capitolato C8 - JMAP: il nuovo protocollo per la posta elettronica.....	14
2.8 Capitolato C9 - ChatSQL: creare frasi SQL da linguaggio naturale.....	15

1 Valutazione del capitolato scelto

1.1 Capitolato C6 - InnovaCity: *Smart city monitoring platform*

1.1.1 Descrizione

- **Proponente:** Sync Lab
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** realizzare una piattaforma di monitoraggio di una smart city in grado di immagazzinare e analizzare una grande quantità di dati ricevuti in input da diversi dispositivi IoT presenti nel territorio. Questo dovrebbe consentire a chi la usa di avere sotto controllo una serie di informazioni sullo stato di salute della città, permettendo decisioni veloci, efficaci ed un'analisi dei conseguenti effetti.

1.1.2 Dominio applicativo

La soluzione si basa su tecnologie che permettono di gestire il *data streaming processing*, cioè che sono in grado di interpretare, arricchire e rendere persistente la grande mole di dati grezza che viene raccolta dai sensori. Si richiede quindi di sviluppare una piattaforma in grado di generare, simulandoli, i dati provenienti dai sensori, in modo tale che siano realistici e li rappresenti in una serie di dashboard. I dati provenienti dai sensori possono essere di varia natura, qui se ne elencano alcuni **esempi**:

- sensori di temperatura: misurano la temperatura ambientale;
- sensori di polveri sottili: rilevano le particelle di polveri sottili nell'aria;
- sensori di precipitazione: misurano la quantità di pioggia o neve caduta;
- sensori di umidità: monitorano il livello di umidità;
- sensori per il livello dei corsi d'acqua: rilevano il livello di acqua nella zona di installazione;
- sensori di guasti elettrici: segnalano l'eventuale assenza di energia elettrica in una determinata zona;
- sensori di soglia: controllano lo stato di riempimento dei contenitori nelle isole ecologiche.

L'applicativo, nella sua componente grafica dovrà fornire una maschera contenente la mappa della città con le informazioni raccolte, oltre che le componenti che permettono di avere un quadro più completo della situazione, come grafici e widget statistici.

1.1.3 Dominio tecnologico

Sebbene la Proponente prediliga non imporre tecnologie specifiche, suggerisce le seguenti in ambito *big data*:

- utilizzo di un framework per la simulazione dei dati, quanto più realistica possibile: si suggerisce il linguaggio *Python* ed eventualmente librerie per la generazione di dati (*Faker* ad esempio);
- utilizzo di un *broker* per disaccoppiare lo stream di dati proveniente dai simulatori: per questo si suggerisce *Apache Kafka*, standard per gestire l'ingresso di dati da molte

fonti;

- database OLAP colonnari come *ClickHouse*: questo software, oltre ad avere una semplice integrazione con *Kafka*, fa parte di una famiglia di database che permette di analizzare con efficienza una grande mole di dati;
- utilizzo di una piattaforma di data visualization come ad esempio *Grafana*.

1.1.4 Sistema di votazione e motivazione della scelta

Il gruppo ha unanimemente deciso di eseguire una votazione su uno *spreadsheet* creato appositamente con lo scopo di delineare i capitoli di maggior interesse. Dopo che ciascun componente ha espresso il proprio livello di interesse nei confronti di ogni capitolo utilizzando una scala da 1 a 5, i risultati della votazione sono stati i seguenti:

Capitolati	Risultato Finale
C1 Knowledge AI	23
C2 Raccomandazione	18
C3 Easy Meal	14
C4 Nuvolaris	12
C5 Warehouse 3D	21
C6 InnovaCity	27
C7 ChatGpt vs BedRock	24
C8 JMAP	9
C9 ChatSQL	21

Il capitolo InnovaCity di SyncLab, avendo ottenuto un punteggio di **27/30**, è stato giudicato il più interessante da perseguire. Tra gli aspetti positivi evidenziati da una discussione di gruppo sono emersi:

- SyncLab è un'azienda composta da più di 200 dipendenti che offre ampio supporto lungo tutto lo svolgimento del progetto; in particolare, i referenti di SyncLab mettono a disposizione delle vere e proprie sessioni di *deep dive* per formare gli studenti sulle tecnologie necessarie per costruire l'architettura proposta (principalmente Kafka e ClickHouse ma non solo). Si ritiene che un supporto di questo genere, mirato ad accrescere le competenze tecniche del gruppo, possa accelerare significativamente il processo di sviluppo;
- Presentazione convincente e di qualità in termini di competenza e chiarezza;
- Incontro conoscitivo molto positivo. La Proponente ha risposto in modo esaustivo alle domande del gruppo;
- Il progetto sarebbe di proprietà esclusiva del gruppo S.W.A.T. Engineering, con la prospettiva di poterlo rilasciare come un progetto open-source in futuro;
- La tecnologia da adottare per la costruzione del framework di simulazione realistica dei dati è libera e la Proponente ha mostrato un atteggiamento positivo nei confronti di possibili sperimentazioni.

1.1.5 Conclusioni

Il capitolato ha suscitato un notevole interesse da parte di tutti i membri del gruppo, principalmente a causa dell'opportunità di applicare in modo concreto e realistico l'architettura di *data streaming* proposta. Questa implementazione richiede l'adozione di tecnologie stimolanti con cui nessun membro del gruppo ha avuto esperienza precedentemente, aggiungendo un elemento di entusiasmo all'approccio al progetto. Inoltre, la Proponente ha dimostrato un forte impegno e ha offerto un ampio supporto tecnico, consolidando ulteriormente la decisione iniziale.

2 Valutazione dei capitolati rimanenti

2.1 Capitolato C1 - *Knowledge management AI*

2.1.1 Descrizione

- **Proponente:** AzzurroDigitale
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** Creare una piattaforma web che consenta di gestire la propria documentazione e di interagire con un chatbot in grado di rispondere in modo specifico ad eventuali domande sui documenti.

2.1.2 Dominio applicativo

Questo progetto unisce il campo del *Knowledge Management* con l'applicazione dell'IA per migliorare la gestione dei documenti e delle conoscenze all'interno di qualsiasi organizzazione. L'applicazione sviluppata offre agli utenti la possibilità di eseguire diverse operazioni sui documenti, tra cui il caricamento, la consultazione e l'eventuale eliminazione. I documenti vengono suddivisi nelle loro singole pagine, trasformati in rappresentazioni vettoriali e archiviati in un database dedicato. Questo approccio consente all'interfaccia chat, che interagisce con un Linguaggio di Modellizzazione del Linguaggio (LLM) a scelta, di sfruttare efficacemente il database e di rispondere in modo intelligente alle richieste degli utenti.

2.1.3 Dominio tecnologico

La Proponente ha incluso all'interno del capitolato alcune tecnologie che dovrebbero facilitare lo sviluppo dell'applicazione, in particolare:

- OpenAi: motore per le funzionalità di NPL;
- LangChain: progetto open-source che consente di utilizzare modelli di IA come *blackbox* senza richiedere una conoscenza dettagliata dei loro funzionamenti interni;
- Chroma: database vettoriale che consente di memorizzare i dati sotto forma di *embeddings* e operare ricerche sfruttando algoritmi quali *Nearest Neighbors* (NN);
- NodeJs e Angular: Node.js e Angular vengono solitamente utilizzati insieme in un'architettura *full-stack* per la creazione di applicazioni web.

2.1.4 Aspetti positivi

- Tra i risvolti del progetto vi è l'aumento della sicurezza sul lavoro, specialmente nel mondo delle fabbriche permettendo agli addetti un rapido e sicuro accesso a documenti e manuali;
- Il progetto risulta utile per qualunque contesto in cui si abbia la necessità di estrarre informazioni da un quantitativo significativo di materiale;
- La Proponente offre un supporto discreto e suggerisce l'utilizzo della metodologia *agile*;

- Presentazione ben realizzata, nonostante non siano state specificate informazioni aggiuntive rispetto a quanto riportato nel capitolato.

2.1.5 Fattori critici

- La Proponente non si è espressa sulla proprietà del codice prodotto e sull'eventuale possibilità di rendere pubblico il *repository* git messo a disposizione;
- L'account premium di ChatGPT utilizzato durante la realizzazione del progetto non sarebbe più a disposizione dopo il completamento dello stesso e questo porterebbe il prodotto finale a non essere dimostrabile in futuro.

2.1.6 Conclusioni

Il capitolato è stato tenuto in considerazione come possibile terza scelta durante la votazione interna dei capitolati: l'ambito ha suscitato curiosità, vista la recente crescita esponenziale di interesse verso i LLM, ma il dominio applicativo non ha convinto il gruppo, una volta messo a confronto con altre proposte.

2.2 Capitolato C2 - Sistemi di raccomandazione

2.2.1 Descrizione

- **Proponente:** Ergon Informatica Srl
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** Sviluppare un sistema di raccomandazione mirato a ottimizzare le attività di marketing e commerciali dell'azienda. Questo sistema consiglierà in modo proattivo quali clienti dovrebbero essere presi in considerazione per specifiche iniziative, identificando i clienti target ideali per tali attività.

2.2.2 Dominio applicativo

Il sistema userà metodologie di Machine Learning, come il K-Nearest-Neighbour, per affrontare un problema specifico attraverso l'applicazione di sistemi di raccomandazione. Quest'ultimo andrà a raccogliere e analizzare i feedback degli utenti relativi ai loro interessi al fine di generare raccomandazioni personalizzate. I feedback possono essere espliciti, esempio punteggio di gradimento, oppure impliciti, come gli elementi acquistati o il tempo trascorso su una specifica pagina. Ci sono 2 modalità di implementazione:

1. **Collaborative Filtering** : valuta la similarità tra gli utenti;
2. **Content-Base Filtering** : valuta la similarità tra i prodotti.

2.2.3 Dominio tecnologico

- Un database: la Proponente da ampia libertà su quale DBMS utilizzare;

- *Surprise*: una libreria Python che consente la costruzione e l'analisi di sistemi di raccomandazione che trattano dati di valutazione espliciti, ad esempio un punteggio di gradimento (da 1 a 5);
- *ML.Net*: una libreria scritta in C#, per il machine learning;
- *.Net*: per sviluppare l'interfaccia grafica desktop oppure web-based.

2.2.4 Aspetti positivi

- La Proponente sembra avere molta esperienza sul campo e fornisce 2 riferimenti aziendali;
- Lo sviluppo è "da zero", non si usano API di terze parti.

2.2.5 Fattori critici

- Scelta dei linguaggi forzata tra C#, che funziona molto bene su ambienti Windows ma su Linux ha problemi di compatibilità, e Python;
- Realizzazione di un algoritmo di supporto alla raccomandazione "personalizzato" che integri più strategie.

2.2.6 Conclusioni

Il capitolato in questione è stato scartato in quanto non offre particolari spunti per l'utilizzo di nuove tecnologie. In particolare poi si nota un'enfasi sulle tecnologie basate sul framework .NET ed il linguaggio C#, le quali sono di scarso interesse per il gruppo.

2.3 Capitolato C3 - *Easy meal*

2.3.1 Descrizione

- **Proponente:** Imola Informatica
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** Creare una piattaforma per migliorare l'efficienza delle prenotazioni e ordinazioni nei ristoranti andando a minimizzare lo spreco di risorse alimentari.

2.3.2 Dominio applicativo

EasyMeal mira a semplificare e migliorare l'esperienza dei clienti e dei ristoratori nei ristoranti, offrendo una piattaforma ove è possibile compiere diverse operazioni:

1. Registrazione di un utente;
2. Prenotazione tavoli;
3. Ordinazione;
4. Interazione con lo staff tramite una chat integrata;

5. Divisione del conto secondo varie possibilità;
6. Consultazione delle prenotazioni;
7. Inserimento di recensioni.

2.3.3 Dominio tecnologico

Si tratta di andare a creare una web-app responsive che permetta di effettuare tutte le operazioni di cui sopra. Il servizio deve essere accessibile sia da PC che da mobile (Android o Ios).

2.3.4 Aspetti positivi

- Il servizio risulterebbe molto utile nel mondo della ristorazione in quanto consentirebbe una migliore gestione delle risorse economiche e culinarie, oltre ad una maggiore efficienza nelle prenotazioni;
- Scelta libera per le tecnologie da usare;
- Supporto offerto una volta a settimana, su prenotazione.

2.3.5 Fattori critici

- Il progetto non è altro che un'applicazione CRUD (Create, Remove, Update, Delete) resa complessa da vari requisiti relativi al dominio dell'ordinazione.

2.3.6 Conclusioni

Nonostante il progetto sembri relativamente semplice da realizzare, il capitolato non ha suscitato un forte interesse all'interno del gruppo. Ciò è principalmente dovuto al fatto che diversi membri hanno già esperienza in progetti simili svolti in passato. Di conseguenza, il gruppo ha, in modo condiviso, deciso di focalizzarsi su capitolati che presentano sfide e opportunità di apprendimento più stimolanti.

2.4 Capitolato C4 - *A ChatGPT plugin with Nuvolaris*

2.4.1 Descrizione

- **Proponente:** Nuvolaris
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** Creare un plugin per ChatGPT usando l'ambiente serverless di Nuvolaris

2.4.2 Dominio applicativo

I plugin per ChatGPT sono componenti aggiuntivi progettati per estendere le abilità dello stesso modello linguistico. Aggiungendo tali plugin, l'utente ha modo di accedere a funzionalità esterne tramite una chat.

Viene richiesta la creazione di un plugin ChatGPT che interagendo direttamente con l'ambiente Nuvolaris dia la possibilità all'utente di generare e modificare una certa categoria di applicativi.

2.4.3 Dominio tecnologico

L'unico vincolo direttamente imposto è quello dell'utilizzo dell'ambiente serverless Nuvolaris. Nuvolaris fornisce un ambiente serverless, il che significa che semplifica la gestione delle risorse di server, consentendo ai programmatori di concentrarsi sulla scrittura del codice dell'applicazione senza preoccuparsi dell'infrastruttura sottostante. Inoltre, risulta necessario interagire con le API messe a disposizione da OpenAI. Per quanto riguarda la natura delle applicazioni generate dal plugin, la scelta delle tecnologie è libera.

2.4.4 Aspetti positivi

- Le tecnologie vincolanti sono interessanti e hanno largo impiego nel mondo del lavoro;
- La tipologia di applicazioni che il plugin dovrebbe implementare non è definita, viene lasciato spazio alla creatività.

2.4.5 Fattori critici

- Il supporto offerto, tramite *Slack*, non sembra essere dei migliori;
- Le informazioni trovate online riguardo la tecnologia Nuvolaris sono scarse. In caso di difficoltà nell'apprenderne l'utilizzo, il gruppo sarebbe vincolato a richiedere il supporto della Proponente (soprattutto nelle fasi iniziali di sviluppo);
- Presentazione frettolosa e mal congegnata.

2.4.6 Conclusioni

Nonostante il capitolato sia sembrato inizialmente molto interessante ai più, non è stato possibile accettare gli aspetti critici individuati. Si ritiene che in un progetto di queste dimensioni la disponibilità da parte della Proponente sia comunque una componente fondamentale.

2.5 Capitolato C5 - WMS3: *warehouse management 3D*

2.5.1 Descrizione

- **Proponente:** San Marco Informatica
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** Creare un magazzino virtuale per agevolare la dislocazione dei vari materiali, controllarne gli spostamenti e gestire i processi di spedizione per ottimizzare gli spazi fisici del magazzino.

2.5.2 Dominio applicativo

Software di questo genere si prestano molto bene in realtà industriali emergenti che vogliano ottimizzare la gestione del magazzino per raggiungere un'elevata efficienza nei flussi. Attraverso l'uso di un ambiente virtuale 3D si possono presentare vantaggi in diverse aree per esempio:

1. La comprensione spaziale, in quanto gli oggetti verrebbero rappresentati in 3 dimensioni;
2. Una migliore visualizzazione delle merci stoccate e il loro tracciamento in *real time*, in caso di spostamenti;
3. L'ambiente dinamico permetterebbe di simulare e identificare eventuali criticità in caso di modifiche strutturali.

2.5.3 Dominio tecnologico

Si tratta essenzialmente di sviluppare un ambiente in 3D usando a nostra discrezione una tra le seguenti tecnologie:

- *Three.js*: una libreria JavaScript cross-browser utilizzata per creare e visualizzare grafica computerizzata 3D animata in un browser Web;
- *Unity*: motore grafico per la creazione di videogiochi e animazioni in 3D;
- *Unreal Engine*: un altro esempio di motore grafico comunemente utilizzato per la creazione di videogiochi.

2.5.4 Aspetti positivi

- La Proponente prevede la possibilità di ricontattare i partecipanti per assunzioni future;
- La Proponente offre ben 3 referenti di progetto.

2.5.5 Fattori critici

- Le tecnologie grafiche come Unreal Engine o Three.js possono risultare molto complesse da padroneggiare, ma nel nostro gruppo manca l'interesse necessario per giustificare uno studio approfondito di tali tecnologie.

2.5.6 Conclusioni

Nonostante il progetto sia molto interessante, in quanto effettivamente utile nel mondo della logistica e dei grandi magazzini, il gruppo non l'ha preso in considerazione a causa dei fattori critici sopra esposti.

2.6 Capitolato C7 - *ChatGPT vs BedRock developer analysis*

2.6.1 Descrizione

- **Proponente:** Zero12
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** L'obiettivo principale del progetto è sviluppare un middleware capace di tradurre i requisiti aziendali in user stories, sfruttando ChatGPT o BedRock. In aggiunta, si prevede la realizzazione di plugin dedicati per ambienti di sviluppo come *Visual Studio* e *Xcode*. Un elemento chiave del progetto riguarda il confronto tra le capacità di ChatGPT e BedRock nel processo di traduzione dei requisiti.

2.6.2 Dominio applicativo

Il progetto si concentra sul dominio dell'automazione delle fasi di sviluppo software e sulla gestione dei requisiti aziendali. Gli obiettivi principali includono la generazione automatica di *user stories* per semplificare la documentazione dei requisiti software e accelerare il processo di sviluppo, la creazione di plugin dedicati per due ambienti di sviluppo, la valutazione delle capacità di ChatGPT e AWS BedRock nell'interpretare il codice sorgente e nell'associare le *user stories* generate ai requisiti e l'individuazione di eventuali test non gestiti una volta interpretati i criteri di accettazione delle *user stories* e il codice sorgente analizzato. Complessivamente il dominio applicativo del progetto riguarda la valutazione delle capacità degli strumenti di IA nell'ambito dello sviluppo software.

2.6.3 Dominio tecnologico

La Proponente raccomanda l'uso delle seguenti tecnologie:

- Amazon web services (AWS): una piattaforma che offre servizi di cloud computing;
- *AWS Fargate*: un motore di calcolo serverless per container che consente la creazione di applicazioni senza la necessità di gestire infrastrutture server;
- *MongoDB*: un DBMS non relazionale di tipo NoSql orientato ai documenti;
- *Node.js*: consigliato per la scrittura di API Restful JSON come supporto per il prodotto;
- *Python*: linguaggio di scripting che verrà usato, in questo caso, per la creazione del plugin per *Xcode*
- *TypeScript*: linguaggio di scripting che verrà usato per la creazione del plugin per *Visual Studio*.

2.6.4 Aspetti positivi

- La Proponente offrirebbe un account ChatGPT Premium, una formazione iniziale sulle tecnologie di Amazon Web Services, una repository Github, Jira e dei crediti AWS;
- Oltre al supporto offerto, prevedono la possibilità di effettuare delle sessioni tecniche di review;
- Si potrebbero apprendere le tecnologie serverless proprietarie di Amazon.

2.6.5 Fattori critici

- La Proponente non ha offerto particolari spunti o chiarimenti in merito alla strategia da adottare per sviluppare la componente di generazione delle *user stories* e creare la componente di analisi del codice sorgente.

2.6.6 Conclusioni

Nonostante un'attrattiva iniziale (il capitolato è stato preso in considerazione come seconda scelta ipotetica), il gruppo ha avuto difficoltà a chiarire appieno l'obiettivo finale del progetto, che è risultato poco definito e comprensibile.

2.7 Capitolato C8 - JMAP: il nuovo protocollo per la posta elettronica

2.7.1 Descrizione

- **Proponente:** Zextras
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** Estendere le funzionalità di *Carbonio*.

2.7.2 Dominio applicativo

Il progetto consiste nell'integrare il protocollo mail JMAP all'interno del software *Carbonio*. Con questo progetto si vuole investigare la possibilità di integrazione del protocollo, ovvero valutare se possa essere un valido sostituto per i protocolli POP3 e IMAP.

2.7.3 Dominio tecnologico

- Deve essere utilizzato il software *Carbonio*.
- È data libertà di scelta per il linguaggio nella quale verrà implementata l'estensione di JMAP, tuttavia viene preferito Java in quanto è il linguaggio con il quale è stato scritto *Carbonio*.

2.7.4 Aspetti positivi

- È un'opportunità per andare ad implementare un protocollo innovativo nel dominio della posta elettronica;
- Viene data libertà nei linguaggi di programmazione da utilizzare.

2.7.5 Fattori critici

- La Proponente non ha menzionato che tipo di supporto offrano;
- Si deve utilizzare *Carbonio*, di loro proprietà con conseguente scarsa documentazione online in caso di problemi;

- Non risulta accattivante dal punto di vista delle tecnologie da apprendere.

2.7.6 Conclusioni

Il progetto non ha suscitato nessun particolare entusiasmo da parte del gruppo per via del tema del capitolato.

2.8 Capitolato C9 - *ChatSQL: creare frasi SQL da linguaggio naturale*

2.8.1 Descrizione

- **Proponente:** Zucchetti Spa
- **Committenti:** Tullio Vardanega e Riccardo Cardin
- **Obiettivo:** Sviluppare un sistema di *prompt* in grado di tradurre una frase in linguaggio naturale in un comando SQL.

2.8.2 Dominio applicativo

In questo progetto si vogliono coinvolgere le capacità dei modelli di linguaggio (LLM) al fine di generare codice SQL valido a partire da una richiesta posta in linguaggio naturale. Bisogna quindi sviluppare un software che, accettando in input una frase in linguaggio naturale, costruisca il prompt giusto affinché possa essere utilizzato in un modello di linguaggio (in modo automatico o manuale) per la generazione della query SQL corrispondente. È possibile esplorare diverse alternative di modelli di linguaggio, utilizzando modelli molto grandi come ChatGPT o modelli molto più piccoli ma più specializzati in questo dominio come quelli ritrovabili sulla piattaforma [Hugging Face](#).

2.8.3 Dominio tecnologico

La Proponente ha dato ampia libertà sull'uso delle varie tecnologie da usare. Non ci sono particolari vincoli.

2.8.4 Aspetti positivi

- Il progetto sarebbe completamente open source;
- Viene offerta libertà sulle tecnologie da impiegare nello sviluppo del progetto.

2.8.5 Fattori critici

- Si tratta di andare a automatizzare la creazione di un prompt, e di usare modelli di linguaggio già addestrati o di richiamare delle API;
- Non è stata fatta menzione sul tipo di supporto che la Proponente ha intenzione di dare.

2.8.6 Conclusioni

Sebbene il progetto possa avere un impatto positivo e possa risultare utile come strumento di accompagnamento per lo sviluppo di applicazioni che coinvolgono l'uso di un database, esso non risulta abbastanza stimolante nel suo sviluppo per i componenti del gruppo.