



<CodeHex16>

unipd.codehex16@gmail.com

Valutazione dei capitolati

Data 24/10/2024

Versione 0.1.0

Sommario

Valutazioni sui capitolati proposti
con tecnologie richieste e considerazioni

Presenze

Matteo Bazzan	Redattore
Luca Ribon	Verificatore
Francesco Fragonas	Verificatore
Gabriele Magnelli	Verificatore
Filippo Sabbadin	Verificatore

Registro delle Versioni

Versione	Data	Autore	Cambiamenti
0.1.0	2024-10-24		Prima stesura

Indice

1. Scopo del documento	1
2. Capitolo scelto	1
2.1. C7 - Assistente Virtuale Ergon	1
2.1.1. Descrizione	1
2.1.2. Stack tecnologico	1
2.1.3. Valutazione	1
3. Capitoli non scelti	2
3.1. C1 - Artificial QI Zucchetti	2
3.1.1. Descrizione	2
3.1.2. Stack tecnologico	2
3.1.3. Valutazione	2
3.2. C2 - Vimar GENIALE Vimar	3
3.2.1. Descrizione	3
3.2.2. Stack tecnologico	3
3.2.3. Valutazione	3
3.3. C3 - Automatizzare le routine digitali tramite l'intelligenza generativa Var Group	3
3.3.1. Descrizione	3
3.3.2. Stack tecnologico	3
3.3.3. Valutazione	4
3.4. C4 - NearYou - Smart custom advertising platform SyncLab	4
3.4.1. Descrizione	4
3.4.2. Stack tecnologico	4
3.4.3. Valutazione	4
3.5. C5 - 3Dataviz Sanmarco Informatica	5
3.5.1. Descrizione	5
3.5.2. Stack tecnologico	5
3.5.3. Valutazione	5
3.6. C6 - Sistema di Gestione di un Magazzino Distribuito M31	5
3.6.1. Descrizione	5
3.6.2. Stack tecnologico	5
3.6.3. Valutazione	6
3.7. C8 - Requirement Tracker Plug-in VSCode bluewind	6
3.7.1. Descrizione	6

3.7.2. Stack tecnologico	6
3.7.3. Valutazione	7
3.8. C9 - BuddyBot azzurro digitale	7
3.8.1. Descrizione	7
3.8.2. Stack tecnologico	7
3.8.3. Valutazione	7

1. Scopo del documento

Lo scopo del documento è valutare i vari capitoli, e indicare i motivi che hanno portato alla scelta del capitolo C7

Tutti i capitoli sono consultabili alla pagina Capitoli

2. Capitolo scelto

2.1. C7 - Assistente Virtuale | Ergon

2.1.1. Descrizione

Il capitolo propone la realizzazione di un sistema che prevede un interfaccia mobile in formato *chatbot* che permette all'utente di interagire, tramite linguaggio naturale in forma testuale, con un assistente virtuale. L'assistente virtuale sfrutta un LLM (Large Language Model) che dovrà essere in grado di rispondere alle richieste poste dall'utente basandosi sul contesto aziendale; quest'ultimo viene fornito tramite documenti aziendali in formato .pdf, .docx e .txt.

2.1.2. Stack tecnologico

Le tecnologie necessarie per la realizzazione del progetto sono:

- **Database relazionale** a scelta tra quelli presenti sul mercato, utilizzato per la memorizzazione dei dati (sottoforma di vettori) relativi al contesto aziendale
- **LLM (Large Language Model)** per la costruzione delle risposte da dare all'utente; è stata fornita una lista da cui scegliere, tra questi risaltano BLOOM e Falcon IA in quanto maggiormente avanzati e adatti a contesti multi lingua
- **API REST** per mettere in comunicazione chatbot e il LLM
- **Comunicazione e manipolazione del database** **TODO**
- **Interfaccia utente** tramite app mobile scegliendo tra il framework .NET MAUI o le tecnologie Android native

2.1.3. Valutazione

Il gruppo ha concordato di assegnare la massima priorità al capitolo n. 7. Questo progetto si è distinto per diversi aspetti ritenuti cruciali:

- La rilevanza del progetto in ambito lavorativo: si tratta di una soluzione che potrebbe essere ampiamente richiesta da diverse aziende, soprattutto nell'ottica di sviluppo di assistenti virtuali e chatbot avanzati.
- La forte integrazione di intelligenza artificiale, in particolare con l'uso di modelli linguistici di grandi dimensioni (LLM), elemento chiave che soddisfa l'interesse del gruppo per l'IA.
- La concretezza del progetto, che si presta ad essere applicato in scenari reali e che potrebbe avere un impatto immediato sul mondo aziendale.

TODO: parlare anche un po' dell'azienda

3. Capitolati non scelti

3.1. C1 - Artificial QI | Zucchetti

3.1.1. Descrizione

L'azienda propone di sviluppare un'applicazione effettuare il ranking di diversi LLM basandosi su dei test che misurano le performance per rispondere a delle domande predeterminate. Il progetto prevede la creazione di un'interfaccia user-friendly per caricare, modificare ed eliminare i dati utilizzati per la fase di ranking. È previsto anche l'utilizzo di un sistema esterno attraverso una API Rest secondo lo standard OpenAPI 3.1 da usare come esempio di valutazione.

3.1.2. Stack tecnologico

Le tecnologie necessarie per la realizzazione del progetto sono:

- Utilizzo di LLM
- Utilizzo di una API Rest secondo lo standard OpenAPI 3.1
- Uso di un database per l'archiviazione delle domande da fare al modello e risposte da confrontare

3.1.3. Valutazione

Nonostante l'interesse del gruppo per l'argomento, il capitolato in questione non è stato scelto perché è risultato poco concreto e limitato dal punto di vista di applicazioni nel mondo reale.

3.2. C2 - Vimar GENIALE | Vimar

3.2.1. Descrizione

Viene proposta la creazione di un chatbot messo a disposizione per 24 ore su 24. L'applicativo web fornisce informazioni sui vari dispositivi forniti dall'azienda, ad esempio può aiutare l'utente durante varie fasi nel processo di installazione di un termostato.

3.2.2. Stack tecnologico

- Utilizzo di un modello AI (LLM)
- Infrastruttura con utilizzo della tecnologia container
- Infrastruttura Amazon Web Services (opzionale)

3.2.3. Valutazione

Questo progetto si distingue per la sua somiglianza con il capitolato 7, ma con alcune differenze importanti:

- L'output del sistema non è basato su dati predefiniti, ma viene generato a partire da informazioni estratte direttamente dal sito web di Vimar, il che aggiunge una dimensione dinamica interessante.
- La presenza di un sistema di risposta intelligente, che sfrutta dati aziendali reali e richiede una forte integrazione con le tecnologie web.

3.3. C3 - Automatizzare le routine digitali tramite l'intelligenza generativa | Var Group

3.3.1. Descrizione

Il progetto prevede lo sviluppo di un servizio ad agenti dove gli utenti possono disegnare localmente un workflow sfruttando le API dei software locali e l'intelligenza artificiale in cloud per automatizzare attività quotidiane che l'utente svolge manualmente.

3.3.2. Stack tecnologico

- Sviluppo in cloud (AWS)
- Creazione di interfaccia drag & drop

- Python / C# / TypeScript
- MongoDB (database)
- React (interfacce applicative web)

3.3.3. Valutazione

3.4. C4 - NearYou - Smart custom advertising platform | SyncLab

3.4.1. Descrizione

L'azienda ha proposto di realizzare dei simulatori per creare i dati GPS che simulino i tragitti e posizione attuale degli utenti lungo il percorso in un GPS. Il programma prende informazioni dall'utente in base alle attività recent, ed in base a queste informazioni seleziona cosa suggerire all'utente nella mappa, con un messaggio personalizzato.

3.4.2. Stack tecnologico

- Python per sviluppare il framework per la simulazione dei dati
- Apache Kafka, HiveMQ, RabbitMQ per disaccoppiare lo strea di informazioni provenienti dai simulatori
- LLM
- Database

3.4.3. Valutazione

La seconda scelta del gruppo è caduta sul capitolato "NearYou - Smart custom advertising platform", proposto da SyncLab. Sebbene la parte di intelligenza artificiale sia meno marcata rispetto alla prima opzione, questo capitolato presenta comunque elementi tecnologici di grande interesse:

- Tecnologie legate alla personalizzazione della pubblicità e all'analisi dei dati, con un focus su algoritmi di machine learning per ottimizzare campagne pubblicitarie.
- Potenziale di utilizzo nel marketing digitale, settore in forte espansione, che richiede soluzioni sempre più innovative per raggiungere target specifici.
- Sfida tecnica interessante, che consentirebbe al gruppo di esplorare nuove aree di sviluppo software, pur mantenendo una certa complessità gestibile.

3.5. C5 - 3Dataviz | Sanmarco Informatica

3.5.1. Descrizione

L'azienda propone la creazione di grafici 3D per aiutare ad avere una migliore visibilità dei dati e avere un'idea più veloce sulle quantità dei dati.

3.5.2. Stack tecnologico

Per la parte 3D del progetto, l'azienda propone l'uso delle seguenti librerie Java:

- three.js
- d3js

Propone anche i seguenti framework frontend:

- Angular (Google)
- React (Meta)

3.5.3. Valutazione

3.6. C6 - Sistema di Gestione di un Magazzino Distribuito | M31

3.6.1. Descrizione

Il capitolato propone un sistema di gestione magazzino che ottimizza le scorte automaticamente, sincronizza i dati in tempo reale tra i magazzini, implementa riasortimenti predittivi tramite machine learning e gestisce i conflitti negli aggiornamenti simultanei

3.6.2. Stack tecnologico

Le tecnologie necessarie per la realizzazione del progetto sono:

- Node.js e Nest.js (usando TypeScript come linguaggio) verranno impiegati per lo sviluppo dei microservizi, grazie alla loro capacità di creare applicazioni server-side performanti e modulari.
- Go potrà essere utilizzato per eventuali componenti ad alte prestazioni, come i servizi di sincronizzazione.
- NATS o Apache Kafka per la comunicazione tra i microservizi.
- Il sistema di orchestrazione e gestione centralizzata sarà ospitato su Google Cloud Platform, che offre servizi cloud affidabili e scalabili in particolare Kubernetes.

- Per l'archiviazione dei dati, si farà uso di MongoDB per la memorizzazione di dati non strutturati, mentre PostgreSQL potrà essere impiegato per la persistenza di dati strutturati.
- Redis potrà essere utilizzato come sistema di caching per migliorare le prestazioni e ridurre la latenza.
- Angular e generalmente SPAs (Single Page Applications) per il frontend per fornire agli utenti una esperienza il più possibile simile a quella di un'applicazione desktop.

3.6.3. Valutazione

3.7. C8 - Requirement Tracker Plug-in VSCode | bluewind

3.7.1. Descrizione

Il capitolato propone un plugin di VSCode per il tracciamento automatico dei requisiti nel codice sorgente, con funzionalità di analisi basate su intelligenza artificiale per suggerire miglioramenti sia al codice che ai requisiti, inclusi quelli derivati da documentazione hardware. Il sistema è progettato per essere espandibile con nuove funzionalità.

3.7.2. Stack tecnologico

Per favorire l'estensibilità e la futura evoluzione del plug-in, si incoraggiano le seguenti tecnologie:

- Visual Studio Code Extension API: per costruire un'architettura modulare, che consenta l'aggiunta di nuove funzionalità in maniera semplice.
- API REST: per connettersi a modelli di AI per l'analisi del codice e dei requisiti.
- Python o Node.js: per l'integrazione con le API AI, con un design flessibile che consenta di aggiungere nuovi linguaggi o componenti senza modifiche significative.
- Modelli AI pre-addestrati (come GPT o simili): per analisi semantiche, con la possibilità di integrare facilmente nuovi modelli o algoritmi in futuro.
- Ollama, o una eventuali alternative: per il deployment locale di LLM (opzionale).

3.7.3. Valutazione

3.8. C9 - BuddyBot | azzurrodigitale

3.8.1. Descrizione

Il capitolato propone Buddy Bot: una piattaforma web innovativa che funge da assistente virtuale intelligente, progettata per semplificare e ottimizzare la gestione delle informazioni aziendali. Integrando diverse fonti di dati attraverso API di terze parti e sfruttando le più recenti tecnologie di intelligenza artificiale, offre un'esperienza utente intuitiva consultabile attraverso un'interfaccia chat.

3.8.2. Stack tecnologico

Le tecnologie necessarie per la realizzazione del progetto sono:

- OpenAI: motore per le funzionalità di NPL (comprensione del testo e generazione delle risposte)
- Node/NestJS: framework per lo sviluppo di applicazioni server-side.
- Langchain: progetto open-source che permette di

integrare modelli di AI senza conoscerne i dettagli interni.

- Spring Boot: framework Java per creare applicazioni standalone e pronte per la produzione
- Angular: framework frontend per la costruzione di applicazioni web moderne, dinamiche e scalabili.

3.8.3. Valutazione