



TECH MINDS

Documento di studio dei capitoli

techminds.unipd@gmail.com

Sommario

Studio dei 9 capitoli proposti per il corso di Ingegneria del Software - Anno 2024/2025.

Changelog

Versione	Data	Descrizione	Autore	Verificatore
1.2	05/11/2024	Correzione degli errori tipografici, revisione stilistica di tutto il documento	Salviato Leonardo Vallotto Caterina	Tutino Giuseppe Squarzone Matteo
1.1	05/11/2024	Modifica posizione e struttura changelog	Corradin Samuele	Tutino Giuseppe Vallotto Caterina
1.0	25/10/2024	Prima versione	Il gruppo Tech Minds	Tutino Giuseppe Vallotto Caterina

Indice

1	Introduzione	3
1.1	Scopo del documento	3
1.2	Fonte dei documenti	3
2	Valutazioni	3
2.1	C1 - ArtificialQI (Zucchetti)	3
2.2	C2 - Vimar GENIALE	4
2.3	C3 - Automatizzare le routine digitali (Var Group)	5
2.4	C4 - NearYou (Sync Lab)	7
2.5	C5 - 3Dataviz (Sanmarco Informatica)	8
2.6	C6 - Sistema di gestione di un magazzino distribuito (M31)	9
2.7	C7 - LLM: Assistente virtuale (Ergon)	10
2.8	C8 - Requirement Tracker - plugin VSCode (Bluewind)	11
2.9	C9 - BuddyBot (Azzurro digitale)	12

Lista delle figure

1 Introduzione

1.1 Scopo del documento

Questo documento vuole fornire una panoramica dei capitolati presentati dalle aziende secondo il punto di vista di ciascun componente del gruppo. Dopo le discussioni avvenute, la preferenza unanime è il capitolato 3.

1.2 Fonte dei documenti

I documenti di presentazione dei capitolati sono presenti nel [sito del docente](#).

2 Valutazioni

2.1 C1 - ArtificialQI (Zucchetti)

2.1.1 Descrizione generale

Il proponente richiede di sviluppare un applicativo in grado di valutare le risposte generate da vari LLM, al fine di poter visualizzare i risultati e comprendere il loro comportamento dato un set di domande e risposte. Il tema alla base del capitolato fa riflettere sulla necessità di un modello potente come «ChatGPT-4-turbo», dal numero di parametri molto alto, oppure se modelli con meno parametri siano sufficienti per ottenere dei risultati accettabili.

2.1.2 Funzionalità obbligatorie richieste

- Archiviazione di una lista di domande e delle risposte attese;
- Programma di esecuzione del test che attraverso una API ponga le domande ad un programma esterno e ne registri la risposta;
- Programma di valutazione della correttezza/verosimiglianza delle risposte ricevute;
- Procedura di presentazione dei risultati dell'esecuzione del test;
- I punti precedenti devono essere integrati in un unico sistema che permetta di utilizzarli come parte di un insieme e non forniti come utility separate.

2.1.3 Funzionalità opzionali richieste

- Archiviazione dei risultati dei test;
- Confronto automatico tra run di test diversi;
- Archiviazione di set di domande/risposte;
- Gestione della configurazione delle chiamate alle API esterne.

2.1.4 Tecnologie da utilizzare

Non sono state proposte tecnologie specifiche all'interno della descrizione del capitolato. Tuttavia, in seguito ad un incontro da remoto con il proponente sono stati consigliati l'utilizzo di framework ampiamente diffusi, come React o Svelte, per evitare rallentamenti e problemi di incompatibilità.

2.1.5 Aspetti positivi

- Possibilità di fare ricerca per valutare se la risposta ottenuta dal programma chiamato via API sia effettivamente compatibile con la risposta attesa;
- Approfondire il tema degli LLM, molto diffuso al giorno d'oggi;
- Il progetto ha una parte centrale di esplorazione delle tecnologie.

2.1.6 Aspetti negativi e rischi

- Il punto principale del capitolato è lo sviluppo del programma di valutazione della correttezza / verosimiglianza delle risposte ricevute dei requisiti obbligatori, che a detta dello stesso proponente può essere complicato;

- L'azienda dà disponibilità per 3 appalti per i gruppi del primo lotto e altri 3 gruppi del secondo lotto, potrebbero dunque non essere tempestivi nelle risposte dato l'alto numero di gruppi da seguire.

2.1.7 Aiuto fornito dall'azienda

L'azienda si rende disponibile per incontri in presenza o remoto a seconda della volontà degli studenti. Potrà essere fornito un programma esterno che espone delle API da testare. Il proponente metterà a disposizione, su richiesta del gruppo, delle macchine su cui eseguire i modelli da testare.

2.1.8 Contatti avuti con l'azienda

Il gruppo ha fissato un colloquio informativo, sulla piattaforma Google Meet, per un approfondimento riguardo i temi del capitolato. Il proponente si è prontamente reso disponibile per rispondere a tutte le nostre domande. L'incontro è avvenuto il 21/10/2024 e il verbale è presente nell'[omonima cartella](#).

2.1.9 Conclusione

La proposta dell'azienda Zucchetti non ci aveva colpito dopo la presentazione tenutasi in classe. Tuttavia, in seguito al colloquio di approfondimento, il gruppo ha compreso meglio la finalità del progetto, che è stato rivalutato in maniera positiva. Nonostante questo il gruppo non ritiene questo capitolato pienamente in linea con i propri interessi.

2.2 C2 - Vimar GENIALE

2.2.1 Descrizione generale

Il progetto richiede di creare un applicativo dotato di un'interfaccia che permetta, grazie ad un LLM, di fare richieste in linguaggio naturale riguardanti le specifiche dei prodotti Vimar presenti nel sito web dell'azienda. Lo scopo dell'applicativo è facilitare la ricerca di informazioni da parte dei professionisti del settore e supportarli nell'installazione dei prodotti Vimar.

2.2.2 Funzionalità obbligatorie richieste

L'azienda richiede di realizzare un'infrastruttura cloud-ready che contenga un applicativo web responsive per l'utente finale e un applicativo server per la gestione dei dati, con documentazione annessa. In particolare:

- L'applicativo web:
 - deve essere responsive e funzionare su smartphone, tablet e desktop via browser;
 - deve avere un sistema di conversazione libera in lingua italiana;
 - deve avere un sistema di feedback da parte dell'utente sulla risposta ottenuta;
 - deve avere una sezione protetta dedicata agli amministratori.
- L'applicativo server:
 - deve avere un sistema di estrazione e raccolta dati dal sito web dell'azienda (obbligatori prodotti impianto Smart e Domotico);
 - deve avere un sistema di indicizzazione dei dati raccolti;
 - deve avere un componente di interrogazione che si interfacci con il sistema di indicizzazione e il modello AI (LLM open source).

L'infrastruttura cloud-ready:

- deve avere un componente per l'applicativo web che si interfacci con l'applicativo server;
- deve utilizzare la tecnologia container;
- deve essere realizzata con il principio IaC (Infrastructure as Code).

L'azienda inoltre richiede l'implementazione di test di unità, d'integrazione ed end-to-end basati sui requisiti, con una copertura almeno del 75% per i primi due e almeno dell'80% per il terzo.

2.2.3 Funzionalità opzionali richieste

Per l'applicativo web è opzionale implementare un sistema di conversazione guidata, un sistema di suggerimenti per domande successive e la visione delle fonti usate per le risposte attraverso dei link di riferimento. Nell'applicativo server si possono aggiungere un controllo sull'output del componente di interrogazione e la possibilità di estendere il database con i prodotti per un impianto tradizionale. Una caratteristica opzionale per l'infrastruttura è la sua realizzazione su AWS.

2.2.4 Tecnologie da utilizzare

- Docker per l'infrastruttura cloud;
- Git per il versionamento del repository di lavoro;
- LLM per il componente di interrogazione.

Altri suggerimenti da parte dell'azienda:

- Flask, Angular, VueJS (applicativo web responsive);
- Python;
- Scrapy, OCRmyPDF (estrazione dati sito web);
- PostgreSQL, TimescaleDB, InfluxDB (database);
- Llama 3.1, Mistral, Bert, Phi (LLM open source).

2.2.5 Aspetti positivi

- Buona disponibilità da parte dell'azienda sia in ambito formativo sia nel controllo dell'avanzamento del progetto;
- Capitolato chiaro ed esaustivo;
- Possibilità di interfacciarsi con un ampio insieme di tecnologie.

2.2.6 Aspetti negativi e rischi

- Tema comune ad altri capitolati proposti.

2.2.7 Aiuto fornito dall'azienda

L'azienda fornisce la possibilità di organizzare molti incontri, in particolare un SAL (Stato Avanzamento Lavori) bisettimanale di un'ora, che può diventare settimanale di mezz'ora. Possono essere fissate altre riunioni per approfondire le tecnologie utilizzate nel progetto. Vengono inoltre richiesti almeno due incontri in presenza con il gruppo per la consegna dei materiali e il collaudo della soluzione.

2.2.8 Contatti avuti con l'azienda

L'azienda è stata contattata via email per alcuni dubbi sulle funzionalità presentate nel capitolato, che alla fine si sono rivelati essere solamente altre funzionalità opzionali.

2.2.9 Conclusione

Nonostante i vari aspetti positivi il gruppo ha preferito optare per un capitolato con un tema più ricercato.

2.3 C3 - Automatizzare le routine digitali (Var Group)

2.3.1 Descrizione generale

Vargroup propone di costruire un software che, sfruttando l'intelligenza artificiale generativa, sia in grado di automatizzare alcune delle azioni che gli utenti compiono quotidianamente negli applicativi

che saranno supportati. Un utente può creare un workflow selezionando i blocchi di interesse e collegandoli tramite un'interfaccia drag and drop. Ogni collegamento corrisponde a un'automazione descritta in linguaggio naturale che sarà interpretata dal modello.

2.3.2 Funzionalità obbligatorie richieste

- Creazione di un client per disegnare i flussi di automazione su piattaforma Windows e/o Mac;
- Tale client deve essere dotato di un'interfaccia drag and drop e di un'interfaccia conversazionale per la definizione delle automazioni;
- Creazione di almeno tre blocchi di automazione;
- Creazione di un'infrastruttura cloud su AWS e integrazione con il client;
- Individuazione dei limiti e dei difetti della soluzione sviluppata;
- Documentazione;
- Bug reporting;
- Implementazione di test automatici con una copertura almeno del 70%;
- Sviluppo dell'applicativo con un'architettura modulare per consentire un'estensione futura della documentazione.

2.3.3 Funzionalità opzionali richieste

Non sono richieste esplicitamente funzionalità opzionali.

2.3.4 Tecnologie da utilizzare

Il proponente richiede l'utilizzo delle seguenti tecnologie:

- Sviluppo del client:
 - Python o C# su Windows, Swift e Swift UI su MacOS;
 - MongoDB o altro database locale;
 - React per lo sviluppo di una webapp.
- Sviluppo delle API cloud:
 - NodeJS;
 - Python;
 - Typescript.

2.3.5 Aspetti positivi

- Progetto interessante e che può avere un impatto positivo nelle attività lavorative e nella vita privata;
- Utilizzo di diverse tecnologie molto diffuse;
- Opportunità di formazione in azienda sulle tecnologie impiegate;
- Apprendimento del design thinking tramite una sessione guidata;
- Il progetto è la giusta combinazione fra sviluppo e innovazione;
- Più membri del gruppo sono interessati all'apprendimento di AWS.

2.3.6 Aspetti negativi e rischi

- Progetto esplorativo volto a riconoscere i limiti delle tecnologie in tale ambito;
- L'utilizzo proficuo e consapevole di AWS richiede una buona conoscenza dello strumento.

2.3.7 Aiuto fornito dall'azienda

L'azienda si è resa disponibile a concordare il piano di controllo dello stato di avanzamento del progetto. Inoltre, si è offerta di pianificare un percorso formativo sulle tecnologie più complesse. Vargroup ha pianificato di svolgere un'attività di design thinking con il gruppo e di fornire esempi su come creare i requisiti di business.

2.3.8 Contatti avuti con l'azienda

Il gruppo ha fissato un colloquio informativo, sulla piattaforma Google Meet, per un approfondimento riguardo i temi del capitolato. Il proponente si è prontamente reso disponibile per rispondere a tutte le nostre domande. L'incontro è avvenuto il 24/10/2024 e il verbale con le domande è disponibile nell'[omonima cartella](#).

2.3.9 Conclusione

A prima vista il progetto ha affascinato tutti i membri del gruppo ma, allo stesso tempo, ha sollevato dei dubbi circa la sua complessità. Il colloquio con Vargroup è stato fondamentale sotto questo frangente ed è riuscito a rassicurare il gruppo rispetto alla fattibilità del progetto. In particolare, il gruppo apprezza la volontà dell'azienda di proporre un progetto innovativo ed esplorativo, volto anche a individuare i limiti dell'applicativo da sviluppare.

2.4 C4 - NearYou (Sync Lab)

2.4.1 Descrizione generale

Il proponente richiede la creazione di un sistema software che si occupi della creazione di campagne pubblicitarie mirate, sfruttando un LLM. Per farlo viene chiesto di implementare una piattaforma che riesca a gestire flussi continui di dati di geolocalizzazione e comportamentali degli utenti. Lo scopo è far aumentare il valore della pubblicità percepito per gli inserzionisti, consentendo loro di avere una maggiore precisione nel targeting.

2.4.2 Funzionalità obbligatorie richieste

- Simulatori di dati geospaziali;
- Configurazione del database con storicizzazione dei dati;
- Implementazione di un tool per lo stream processing di più sorgenti;
- Implementazione di un LLM per generare gli annunci;
- Sviluppo di una dashboard per visualizzare i dati;
- Web-app che simula un utente che segue un percorso e riceve gli annunci pubblicitari;
- Test che dimostrino il corretto funzionamento dei servizi;
- Presenza di documentazione.

2.4.3 Tecnologie da utilizzare

Il proponente non ha imposto vincoli stretti sulle tecnologie da usare, ma ha solamente elencato i tipi di software che è necessario utilizzare:

- Framework per la simulazione dei dati;
- Broker per gestire lo stream come MQTT;
- Strumento per lo stream processing che preleva i dati per consegnarli al LLM;
- Strumento per processare le richieste tramite LLM;
- Database capace di elaborare dati geospaziali;
- Strumenti per la data visualization come Grafana;

2.4.4 Aspetti positivi

- Uso di tecnologie non ancora studiate come lo stream processing e l'elaborazione di dati geospaziali;
- Il proponente si rende disponibile a incontri, anche formativi, per guidare il gruppo.

2.4.5 Aspetti negativi e rischi

- Il progetto vuole fare uso di molte tecnologie, costringendo a uno studio superficiale di tutte;
- Il progetto ha una complessità molto alta e richiede la consegna di sette componenti. È molto probabile che non si riesca a completarlo nel tempo a disposizione;
- L'argomento del progetto non stimola l'interesse dei componenti del gruppo.

2.4.6 Aiuto fornito dall'azienda

L'azienda propone degli incontri di revisione dello stato di avanzamento. In particolare possono essere del tipo:

- Analisi dei requisiti;
- Formazione;
- Verifica dello stato di avanzamento dei lavori;
- Dimostrazione di una live demo del prodotto.

2.4.7 Contatti avuti con l'azienda

L'azienda è stata contattata via email per richiedere chiarimenti sulle modalità di presentazione degli annunci, in particolare se fosse necessario sviluppare una web-app. Ci è stato risposto che il focus principale del progetto non risiede nella modalità di visualizzazione, sebbene questa sarà cruciale nel prodotto finale. In ogni caso, l'azienda propone di concordare l'utilizzo di applicativi che permettano una visualizzazione pronta all'uso.

2.4.8 Conclusione

Abbiamo apprezzato l'impegno di cui si fa carico SyncLab per formare i componenti del gruppo, tuttavia riteniamo che la proposta non sia adatta a noi dati gli aspetti negativi evidenziati nella sezione 2.4.5.

2.5 C5 - 3Dataviz (Sanmarco Informatica)

2.5.1 Descrizione generale

Il progetto chiede di realizzare un'interfaccia web per la visualizzazione in forma tridimensionale di dati tramite barre verticali (istogramma 3D) con i relativi dati di origine in forma tabulare.

2.5.2 Funzionalità obbligatorie richieste

- Il grafico dovrà avere le funzioni classiche di un ambiente 3D (rotation, zoom, pan, auto-positioning);
- Possibilità di selezionare un elemento del grafico (barra) o una cella della griglia e di nascondere / opacizzare le barre con i valori superiori o inferiori al valore della barra selezionata;
- Lasciare come unici elementi visibili / non opacizzati i top X / bottom Y valori;
- Possibilità di visualizzare il piano parallelo alla base, che rappresenta il valore medio globale e nascondere / opacizzare le barre con i valori superiori o inferiori al valore medio;
- Visualizzare i valori corrispondenti alla barra.

2.5.3 Funzionalità opzionali richieste

- Poter visualizzare il piano parallelo alla base che rappresenti il valore medio di un singolo elemento dell'asse (x o y);

2.5.4 Tecnologie da utilizzare

- Three.js;
- D3.js;

- Angular;
- React.

2.5.5 Aspetti positivi

- Tema particolare rispetto agli altri capitolati proposti;
- Unico capitolato che tratta il 3D.

2.5.6 Aspetti negativi e rischi

- Poco supporto nella fase di sviluppo da parte dell'azienda;
- Le conoscenze da acquisire vengono considerate dal gruppo meno interessanti rispetto a quelle degli altri capitolati.

2.5.7 Aiuto fornito dall'azienda

L'azienda fornirà un supporto maggiore nelle fasi di analisi dei requisiti e progettazione mentre durante la fase di implementazione sarà disponibile solo in caso di eventuali difficoltà. In generale verranno fissati degli incontri periodici per monitorare l'avanzamento del progetto.

2.5.8 Contatti avuti con l'azienda

Il gruppo ha richiesto un colloquio informativo da remoto, su piattaforma Google Meet. Il verbale relativo all'incontro del 18/10/2024 è presente nell'[omonima cartella](#) nella repository.

2.5.9 Conclusione

Sebbene a primo impatto il progetto abbia colto l'interesse generale dei membri, il gruppo ha optato per un capitolato differente in seguito ai colloqui avuti con le aziende. Tale scelta è stata dettata dal fatto che le tecnologie da utilizzare in altri capitolati sono risultate più interessanti e utili a scopo didattico.

2.6 C6 - Sistema di gestione di un magazzino distribuito (M31)

2.6.1 Descrizione generale

Il progetto si propone di sviluppare un sistema di gestione distribuito per una rete di magazzini, con l'obiettivo di ottimizzare i livelli di scorte e minimizzare i tempi di risposta. L'architettura del sistema sarà basata su microservizi, garantendo operatività indipendente per ciascun magazzino e sincronizzazione in tempo reale con un sistema centrale.

2.6.2 Funzionalità obbligatorie richieste

- Sincronizzazione in tempo reale dei dati di inventario tra i magazzini e il sistema centrale;
- Gestione autonoma dei magazzini tramite microservizi;
- Risoluzione dei conflitti negli aggiornamenti concorrenti;
- Riassortimento predittivo basato su machine learning;
- Monitoraggio centralizzato e reportistica;
- Gestione delle operazioni di trasferimento tra magazzini.

2.6.3 Funzionalità opzionali richieste

- Protezione dei dati e della confidenzialità delle comunicazioni;
- Monitoraggio del sistema.

2.6.4 Tecnologie da utilizzare

- Node.js, Nest.js (TypeScript) per lo sviluppo di microservizi;
- Go per componenti ad alte prestazioni;
- NATS o Apache Kafka per la gestione della comunicazione tra i microservizi;

- Google Cloud Platform, Kubernetes per l'orchestrazione;
- MongoDB per dati non strutturati, PostgreSQL per dati strutturati;
- Redis per il caching;
- Angular per l'interfaccia utente.

2.6.5 Aspetti positivi

- Il progetto è interessante perché propone un modo innovativo di gestire un complesso di magazzini;
- Machine learning utilizzato per un obiettivo diverso da un banale ChatBot;
- Supporto per il riassortimento predittivo tramite machine learning.

2.6.6 Aspetti negativi e rischi

- L'uso di molte tecnologie può portare ad una conoscenza superficiale di ognuna o a uno studio approfondito, con conseguente probabilità di una consegna in ritardo.

2.6.7 Aiuto fornito dall'azienda

M31 fornirà supporto tecnico e metterà a disposizione un esperto per guidare l'analisi e lo sviluppo delle soluzioni. L'azienda supervisionerà la fase di analisi dello stato dell'arte.

2.6.8 Conclusione

Anche se questo capitolato ha suscitato l'interesse di qualche componente del team, non è stato scelto in quanto altri sono stati ritenuti migliori.

2.7 C7 - LLM: Assistente virtuale (Ergon)

2.7.1 Descrizione generale

Il progetto riguarda lo sviluppo di un Assistente Virtuale basato su LLM (Large Language Models) per assistere i clienti nella ricerca di informazioni sui prodotti aziendali. L'obiettivo è automatizzare l'interazione cliente-produttore, fornendo risposte rapide e accurate su prodotti disponibili, riducendo la necessità di contattare specialisti umani per ogni richiesta.

2.7.2 Funzionalità obbligatorie richieste

- Database relazionale per la gestione dei dati;
- Modello LLM con API per l'interrogazione;
- Interfaccia utente per la configurazione della piattaforma;
- Interfaccia utente per l'interazione tra IA e Utente;
- Elementi indipendenti ma in grado di dialogare tra loro;
- Comunicazione con il database (vedi tecnologie da utilizzare);
- Sviluppo di un'app mobile;
- Sviluppo del backend Web per la configurazione della piattaforma;
- Feedback da parte dell'utente;
- Configurazione da backend dei template relativi alle domande base degli utenti e fornire dei template di risposta;
- Documentazione.

2.7.3 Funzionalità opzionali richieste

Approfondire il feedback dell'utente con ulteriori domande.

2.7.4 Tecnologie da utilizzare

- Sql Server Express, MySql o MariaDB;
- BLOOM;
- Falcon IA;
- Pythia;
- Italia by iGenius;
- Minerva;
- API REST;
- Connettori standard fonte dati ODBC;
- Middleware (JSON) per gestire comunicazione tra i componenti;
- .NET MAUI;
- Android.

2.7.5 Aspetti positivi

- Offerta di corsi per l'introduzione a due tecnologie da utilizzare;
- Il progetto dà una panoramica completa dello sviluppo software perchè chiede anche di gestire dati, integrare con API REST, sviluppare interfaccia utente e gestire la connessione tra tutti questi elementi.

2.7.6 Aspetti negativi e rischi

- Dato che il modello richiede una fase di miglioramento continuo basata sui feedback degli utenti, c'è il rischio che non ci siano i tempi necessari per fornire un buon risultato;
- Tema comune ad altri capitolati proposti.

2.7.7 Aiuto fornito dall'azienda

- Supporto da parte del team di R&D in varie fasi di sviluppo (non specificate);
- Interazione da remoto o in presenza;
- Due corsi online sui sistemi LLM e su .NET MAUI.

2.7.8 Conclusione

La proposta risulta meno interessante rispetto agli altri capitolati che riguardano un assistente virtuale.

2.8 C8 - Requirement Tracker - plugin VSCode (Bluewind)

2.8.1 Descrizione generale

Requirement tracker è un plugin per VS Code che attraverso l'uso di intelligenza artificiale analizza il codice, i requisiti (inclusi manuali e datasheet di componenti hardware) e automatizza il tracciamento di essi nel codice sorgente per fornire suggerimenti, migliorandone quindi la qualità.

2.8.2 Funzionalità obbligatorie richieste

- Architettura estensibile;
- Caricamento e scansione dei sorgenti per identificare i requisiti;
- Configurabilità del sistema per lavorare con i requisiti derivati dai vari documenti;
- Supporto ai linguaggi C/C++;
- Visualizzazione dei risultati con la possibilità di navigare e filtrarli per nome del requisito, file o sezione del codice;
- Integrazione con API per inviare porzioni di codice e requisiti;
- Suggerimenti per la scrittura dei requisiti, come:

- Rendere i requisiti più chiari e specifici;
- Identificare informazioni tecniche mancanti;
- Riformulare i requisiti per renderli conformi alle best practice di scrittura tecnica.

2.8.3 Funzionalità opzionali richieste

- Supporto per più linguaggi, ad esempio Rust;
- Visualizzazione grafica avanzata;
- Valutazione della compatibilità dello strumento sviluppato con le specifiche per la scrittura dei requisiti dettate dalla normativa in uso.

2.8.4 Tecnologie da utilizzare

- VScode Extension API;
- API REST;
- Python o Node.js;
- Modelli AI pre-addestrati (GPT e simili);
- Ollama o simili.

2.8.5 Aspetti positivi

Progetto che si distingue dalle altre proposte dato che ha l'obiettivo di creare un'estensione di Visual Studio Code.

2.8.6 Aspetti negativi e rischi

- Esposizione poco chiara del capitolato;
- Si reputa di difficile fattibilità la realizzazione del progetto.

2.8.7 Aiuto fornito dall'azienda

L'azienda si rende disponibile per incontri da remoto per aggiornamenti sul progresso e sessioni di consulenza. Inoltre, possono essere concordati incontri in presenza per discussioni più dettagliate, revisioni approfondite e supporto pratico.

2.8.8 Conclusione

L'argomento proposto dal capitolato non è di interesse per il gruppo che, anche per i motivi elencati nella sezione 2.8.6, non ha ritenuto necessario contattare l'azienda per un colloquio di approfondimento.

2.9 C9 - BuddyBot (Azzurrodigitale)

2.9.1 Descrizione generale

BuddyBot è un assistente virtuale che aggrega informazioni da GitHub, Confluence, Jira, Slack e Telegram, utilizzando tecnologie di Intelligenza Artificiale per migliorare l'efficacia e la personalizzazione del supporto.

2.9.2 Funzionalità obbligatorie richieste

- Piattaforma web per interfacciarsi al bot;
- Backend che fa uso delle API di Confluence, GitHub, Jira;
- Database per memorizzare le chat degli utenti;
- Test automatici con code coverage adeguata per garantire il corretto funzionamento;
- Documentazione tecnica;
- Sistema di bug reporting.

2.9.3 Funzionalità opzionali richieste

Fonti di informazioni aggiuntive, ovvero canali di comunicazione aziendali come gruppi Slack e Telegram.

2.9.4 Tecnologie da utilizzare

- OpenAI;
- Langchain;
- Angular;
- Node/NestJS;
- Spring Boot.

2.9.5 Aspetti positivi

- Le figure messe a disposizione dall'azienda hanno ruoli e competenze diverse;
- La motivazione e gli obiettivi del progetto sono chiari e ben esposti. Ciò facilita il processo di analisi dei requisiti.

2.9.6 Aspetti negativi e rischi

- Il progetto non sembra molto innovativo visto che esistono già da tempo dei plugin per ChatGPT e altri LLM che permettono di integrarsi con fonti esterne;
- L'applicativo sembra una semplice webapp che fa da wrapper ad un LLM.

2.9.7 Aiuto fornito dall'azienda

L'azienda mette a disposizione quattro figure di riferimento per rispondere alle esigenze del gruppo.

2.9.8 Conclusione

La proposta non è stata ritenuta allo stesso livello della concorrenza visti gli aspetti negativi indicati alla sezione 2.9.6. Ciò ha portato il gruppo a concentrarsi su proposte giudicate più allettanti.