



ERROR_418

DOCUMENTAZIONE PROGETTO

Valutazione Capitolati

Mail:	error418swe@gmail.com
Redattori:	Antonio Oseliero, Alessio Banzato
Verificatori:	Riccardo Carraro, Giovanni Gardin, Rosario Zaccone
Amministratori:	Silvio Nardo, Mattia Todesco
Destinatari:	T. Vardanega, R. Cardin

1 Valutazione Capitolato Scelto, C5 – WMS3: Warehouse Management 3D

1.1 Descrizione

- Proponente:
 - Sanmarco Informatica S.p.a.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obbiettivo:
 - Creare un ambiente 3D che permetta la gestione di un magazzino.

Lo scopo di questo capitolato è quello di fornire un modo nuovo per visualizzare un magazzino tramite il 3D, controllare la posizione di materiali e scaffalature con la possibilità di modificarne la posizione. Il proponente richiede che il programma permetta sessioni volatili senza persistenza delle modifiche (lo spostamento di un elemento nel magazzino viene gestito tramite una notifica inviata ai magazzinieri che noi applichiamo sotto forma di API in modo che sia poi il richiedente ad integrarla con i suoi sistemi).

1.2 Dominio Tecnologico

L'azienda (all'interno del capitolato e nel meeting privato con il nostro gruppo) lascia molta libertà in merito alle tecnologie da utilizzare. Per quanto riguarda lo sviluppo dell'ambiente 3D suggerisce:

- Three.js;
- Unity;
- Unreal.

Anche per quanto riguarda la scelta del database l'azienda lascia libertà decisionale suggerendo comunque l'uso di un database relazionale.

1.3 Motivazione della Scelta

- L'idea è molto originale, utile e permette di esplorare ambiti poco conosciuti come la modellazione 3D;
- La libreria Three.js risulta molto interessante e moderna, tenendo conto anche del supporto che l'azienda può fornire in quanto specializzata su di essa;
- L'azienda è sembrata molto disponibile, come mostratosi dalle risposte tempestive dateci negli ultimi giorni, difatti metterà a disposizione figure di diverso livello in modo tale da poter rispondere nella maniera più appropriata alle nostre esigenze, come detto durante la giornata di presentazione dei capitolati.

1.4 Conclusioni

Nonostante i dubbi iniziali, la proposta offerta da Sanmarco Informatica S.p.a. ha cominciato sempre di più ad interessarci, complice anche la disponibilità, simpatia e competenza del referente che ci ha offerto subito l'opportunità di un incontro per chiarire i punti non chiari del capitolato. La possibilità di lavorare con un ambiente 3D è stimolante e sono chiare le possibili applicazioni reali.

2 Valutazione C1 – Knowledge Management AI

2.1 Descrizione

- Proponente:
 - azzurro digitale.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obiettivo:
 - Semplificare la consultazione di informazioni all'interno di un'azienda.

Il capitolato punta a semplificare la consultazione di diverse tipologie di informazioni aziendali sfruttando un'intelligenza artificiale, il cui training avverrà grazie API di terze parti. In particolare si richiede lo sviluppo di una piattaforma web all'interno della quale sarà possibile caricare, consultare ed eliminare i documenti (che verranno poi indicizzati) e utilizzare una chat per interagire con il motore di intelligenza artificiale.

2.2 Dominio Tecnologico

L'azienda consiglia alcune tecnologie sia per la parte legata alla creazione della piattaforma web che per la parte di elaborazione dei documenti e API per l'intelligenza artificiale. Le tecnologie consigliate sono le seguenti:

- Node.js;
- Angular;
- OpenAI;
- LangChain.

2.3 Conclusioni

Il capitolato risulta essere interessante per l'adozione di tecnologie innovative e in particolare per l'ampio raggio d'uso del prodotto che propone, in quanto risulta essere molto utile in diversi ambienti di lavoro. Ad esempio potrebbe essere utilizzato sia in una postazione d'ufficio per richiedere informazioni di carattere amministrativo, che in una postazione all'interno di una fabbrica per comprendere il funzionamento di un determinato strumento.

3 Valutazione C2 – Sistemi di Raccomandazione

3.1 Descrizione

- Proponente:
 - Ergon Informatica.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obbiettivo:
 - Utilizzare il machine learning per creare un sistema di raccomandazione.

In questo capitolato si vuole creare un sistema di raccomandazione che usi il machine learning per migliorare le sue funzionalità. Il prodotto sarà composto da:

- Un database che raccoglie tutti i dati relativi al comportamento dei clienti, e quindi anche i prodotti a loro correlati;
- Il sistema di raccomandazione, che utilizza i dati del database;
- Un'interfaccia utente che permetta di visualizzare i primi N prodotti correlati a un dato utente oppure i primi N clienti correlati a un dato prodotto.

Questo sistema dovrà quindi calcolare e stimare le correlazioni tra clienti e prodotti, ma anche tra clienti stessi, in modo da utilizzare queste correlazioni anche per gestire le correlazioni sui prodotti.

3.2 Dominio Tecnologico

Risulta esserci molta scelta per quanto riguarda le tecnologie, dato che l'azienda ne propone diverse lasciando anche la libertà di adottarne altre.

Per quanto riguarda il database vengono consigliati:

- Sql Server Express;
- MySQL;
- MariaDB.

Per il sistema di raccomandazione:

- ML.NET;
- Surprise (libreria Python).

Mentre per l'interazione tra database e applicativo vengono consigliate:

- Entity Framework (ORM), in caso si usi ML.NET;
- ODBC, in caso si usi Surprise;
- Middleware, ad esempio JSON, se si vuole l'indipendenza del sistema dal database.

L'azienda, inoltre, rende possibile la condivisione di un set di dati da usare per l'apprendimento del modello di machine learning.

3.3 Conclusioni

La logica alla base del capitolato è molto complessa, e inoltre le capacità del gruppo risultano essere non allineate. Tutto ciò ha fatto pensare che la scelta di questo capitolato avrebbe portato a una situazione in cui il tempo di studio richiesto per la piena comprensione delle tecnologie e del capitolato in sé avrebbe causato un rallentamento considerevole del ritmo di lavoro.

4 Valutazione C3 – Easy Meal

4.1 Descrizione

- Proponente:
 - Imola Informatica.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obiettivo:
 - Web app per migliorare l'esperienza culinaria nei ristoranti.

Imola Informatica propone lo sviluppo di EasyMeal, una web app innovativa che vuole trasformare il settore dei ristoranti, semplificando la prenotazione e l'esperienza culinaria per gli utenti.

I clienti possono anticipare l'esperienza culinaria creando il proprio ordine da qualsiasi luogo in base alle proprie esigenze, allergie e preferenze alimentari, oltre che specificando l'orario di arrivo nel locale. L'applicazione facilita l'interazione con lo staff del ristorante, consente la divisione del conto tra i partecipanti e contribuisce a ridurre lo spreco alimentare grazie a una pianificazione della spesa più precisa. Includendo funzionalità come la registrazione, la prenotazione di tavoli, l'ordinazione collaborativa dei pasti, l'interazione con il personale del ristorante, la divisione del conto, la consultazione delle prenotazioni da parte dei ristoratori e la possibilità di inserire feedback e recensioni, EasyMeal si propone di offrire una convenienza, personalizzazione ed efficienza superiori sia ai clienti che ai ristoratori.

4.2 Dominio Tecnologico

Il proponente non fornisce tecnologie particolari e lascia libertà di scelta. L'unico vincolo imposto è che venga sviluppata un'applicazione web responsive (PC, iOS e Android).

4.3 Conclusioni

Lo sviluppo di un'applicazione sia per iOS che Android risulta essere molto interessante, e sono state apprezzate la precisione con cui sono stati presentati i requisiti e l'alta disponibilità da parte dell'azienda anche per quanto riguarda gli incontri periodici per monitorare l'avanzamento del progetto. Il numero di richieste minime risulta però essere elevato e rischia di portare a tempi di sviluppo molto lunghi.

5 Valutazione C4 – A ChatGPT plugin with Nuvolaris

5.1 Descrizione

- Proponente:
 - Nuvolaris.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obiettivo:
 - Creare un plugin di ChatGPT usando Nuvolaris serverless.

Il capitolato si propone di ridurre, tramite IA, la barriera in ingresso per la gestione di setup di cloud computing complessi e normalmente riservati ad utenti esperti.

Il capitolato prevede lo studio e l'utilizzo di diverse tecnologie per ottenere:

- Costruzione e utilizzo di plugin di ChatGPT;
- Automatizzazione della costruzione e modifica di applicazioni in base alla richiesta dell'utente;
- Una serie di template da cui poi verranno generate le applicazioni richieste;
- Corretta gestione e modifica automatica dei file di configurazione per la generazione delle applicazioni.

5.2 Dominio Tecnologico

ChatGpt, Nuvolaris, Redis, in base al tipo di applicazione possono variare le tecnologie utilizzate, ad esempio per applicazioni CRUD si può usare SQL.

5.3 Conclusioni

Il capitolato non risulta chiarissimo nella proposta, sono state necessarie delle mail all'azienda per chiarire alcuni punti non chiarissimi che sono arrivate molto celermente. In generale uno dei motivi per cui abbiamo faticato nel capire la proposta di Nuvolaris è che i membri del team non si sono mai trovati a dover interagire con Docker o Kubernetes.

L'azienda mette a disposizione molte risorse quali: account ChatGPT Pro, documentazione, ambiente Nuvolaris dedicato e completo in cloud.

6 Valutazione C6 – SyncCity: Smart city monitoring platform

6.1 Descrizione

- Proponente:
 - SyncLab.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.

- Obbiettivo:
 - Creare una piattaforma che rappresenti in una serie di dashboard dati provenienti da molti sensori per il monitoraggio della qualità della vita di una città.

L'azienda propone lo sviluppo di una piattaforma che rappresenti una serie di dati ricavati da diversi sensori collocati in una città (questi dati devono essere opportunamente simulati o ottenuti da sensori reali) in modo da rappresentarne lo stato di salute. Il capitolato prevede lo studio e l'utilizzo di diverse tecnologie per ottenere:

- Implementazione di Simulatori di dati con documentazione relativa;
- Configurazione del database per lo storage dei dati;
- Piattaforma di stream processing mediante invio di dati a Kafka;
- Sviluppo di una dashboard a fini di consultazione dei dati raccolti mediante Grafana;
- Testing con copertura $\geq 80\%$.

Il proponente nomina come informazioni da rappresentare per esempio:

- Temperatura, espressa in $^{\circ}\text{C}$;
- Polveri sottili, espressa in $\mu\text{g}/\text{mc}$;
- Umidità, espressa in percentuale;
- Livello dell'acqua nella zona d'installazione del sensore;
- Guasti elettrici, 0 o 1 in caso si verifichi un'interruzione della corrente nella zona d'installazione del sensore;
- Riempimento dei vari conferitori di un'isola ecologica, 0 o 1 a seconda se sia piena o meno.

6.2 Dominio Tecnologico

L'azienda suggerisce come tecnologie da impiegare:

- Script Python (o altri linguaggi) e librerie per la generazione dati per ottenere una simulazione dati realistica;
- Per lo stream processing: Apache Kafka;
- Per lo storage dei dati: ClickHouse (database colonnare);
- Per il data visualization: Grafana.

6.3 Considerazioni

Interessante applicazione IoT, la possibilità di modellare dei dati rappresentandoli in un ambiente che mostra lo stato di salute di una città è appassionante e in linea con i bisogni moderni degli utenti sempre più attenti alla salute. I requisiti sono chiari e ben esposti e lo stack tecnologico ben definito ma comunque flessibile, il capitolato è in generale completo ed esaustivo.

7 Valutazione C7 – ChatGPT vs BedRock developer analysis

7.1 Descrizione

- Proponente:
 - Zero12.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obbiettivo:
 - Utilizzare il machine learning per produrre epic e user stories automaticamente.

L'obiettivo prefissato dal capitolato è la realizzazione di un applicativo in grado di analizzare i requisiti di business e il codice sorgente al fine di produrre epic&user stories, mediante l'utilizzo di sistemi come ChatGPT e AWS BedRock. Il capitolato pertanto si concentra sull'utilizzo di sistemi di comprensione e analisi testuali sfruttando le capacità dell'emergente tecnologia dell'intelligenza artificiale. Una volta generate le epic&user storie, la verifica di correttezza e completezza dovrà avvenire manualmente tramite la webapp, dando la possibilità all'utente di visualizzarle e valutarle.

Il capitolato prevede lo studio e l'utilizzo di diverse tecnologie per ottenere:

- Middleware in grado di ricevere in input requisiti di business e codice sorgente per la produzione di epic&user stories;
- Plugin per VisualStudio Code;
- Plugin per Apple XCode;
- Sviluppo modulare dell'applicativo in modo da poter confrontare il sistema utilizzando ChatGPT rispetto a AWS BedRock;
- Architettura basata su micro-servizi.

7.2 Dominio Tecnologico

Tecnologie consigliate:

- Gestione del container: AWS Fargate;
- Per lo storage dei dati: MongoDB;
- Sviluppo di API: NodeJS;
- Linguaggio per lo sviluppo del plugin per XCode: Python;
- Linguaggio per lo sviluppo del plugin per VisualStudio Code: Typescript.

7.3 Conclusioni

Si tratta di un prodotto dedicato agli sviluppatori, difficile da testare il grado di correttezza dell'output, XCode richiede MacOS (ma Zero12 fornirebbe computer nella loro sede), forte carattere esplorativo, sarebbe un primo approccio ad AWS (standard di settore), crediti AWS inclusi.

8 Valutazione C8 – JMAP: il nuovo protocollo per la posta elettronica

8.1 Descrizione

- Proponente:
 - Zextras.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obbiettivo:
 - Utilizzare il protocollo JMAP per realizzare un'applicazione per lo scambio di email.

Il capitolato si propone di esplorare nuovi sviluppi nella comunicazione e-mail, probabilmente la tecnologia di comunicazione più utilizzata al mondo. Per farlo bisogna lavorare con il protocollo JMAP che dovrà sostituire il protocollo IMAP precedentemente utilizzato per queste applicazioni. Scopo di tale lavoro è, da parte del proponente, capire se ha senso investire tempo e denaro per estendere questo standard in Carbonio, un servizio di collaborazione che offre un'insieme di funzionalità con focus principale sulla gestione delle email.

È previsto lo sviluppo di un servizio che permetta:

- Invio e ricezione di mail;
- Gestione, eliminazione, condivisione di una cartella;
- Gestione dei contenuti delle cartelle.

Opzionalmente anche l'implementazione di un sistema di sincronizzazione che permetta ad un client di mantenersi aggiornato con gli ultimi aggiornamenti della casella di posta visualizzata, contenente anche Calendari, Rubriche contatti, contatti e appuntamenti. Altri vincoli da rispettare:

- Il servizio sviluppato deve essere eseguibile in un sistema container;
- Il servizio sviluppato deve essere scalabile mediante l'inizializzazione di più nodi stateless;
- Opzionalmente aggiungere stress test che riescano a misurare le performance della soluzione.

8.2 Dominio Tecnologico

Tecnologie consigliate:

- Linguaggio di programmazione: Java;
- Librerie per l'implementazione del protocollo JMAP: iNPUTmice o jmap;
- Sistema per gestione container: Docker.

8.3 Conclusioni

Capitolato di carattere esplorativo basato su standard recenti, è un lavoro basato sul protocollo e non su applicazione, il quale tenta di innovare il servizio email. La documentazione fornita è estensiva e il supporto è fornito direttamente dal development team di Carbonio. È interessante la possibilità di valutare le performance tramite stress test.

9 Valutazione C9 – ChatSQL: creare frasi SQL da linguaggio naturale

9.1 Presentazione

- Proponente:
 - Zucchetti.
- Committente:
 - Prof. Tullio Vardanega e Prof. Riccardo Cardin.
- Obbiettivo:
 - Sviluppare un'applicazione che permetta di generare prompt di testo utili alla creazione di comandi SQL.

9.2 Descrizione del capitolato

Il proponente chiede di sviluppare un'applicazione che permetta di generare, immettendo una richiesta in linguaggio naturale e il database (o parte di esso) in un modello di machine learning adeguatamente istruito, una frase anch'essa in linguaggio naturale, la quale attraverso l'utilizzo di ChatGPT possa generare i comandi SQL richiesti. L'applicazione deve svolgere quindi i seguenti compiti:

- Archiviazione della descrizione della struttura di un database, possibilmente commentata in tutte le sue parti;
- Maschera di richiesta di una frase di interrogazione del database in linguaggio naturale;
- Procedura che combina la richiesta di interrogazione con le informazioni della struttura del database creando un "prompt" che sottoposto ad un sistema di AI fornisce l'interrogazione equivalente al linguaggio naturale in linguaggio SQL.

9.3 Dominio Tecnologico

Il proponente non pone vincoli sulle tecnologie da utilizzare, ma suggerisce, riguardo i Large Language Model, l'utilizzo di ChatGPT, Palm o LLaMa.

9.4 Conclusioni

Sono emersi dubbi su come valutare i prompt generati e ci si è soffermati su come elaborare il dataset per ricavare le informazioni necessarie alla creazione del prompt. Interessante l'utilizzo di tecnologie attuali come le intelligenze artificiali e la possibilità di fare training di modelli di machine learning, in questo aiuta la flessibilità nella scelta dei modelli di Large Language Model a supporto.