



Code7Crusaders

Software Development Team

Piano di Progetto

Membri del Team:

Enrico Cotti Cottini, Gabriele Di Pietro, Tommaso Diviesti
Francesco Lapenna, Matthew Pan, Eddy Pinarello, Filippo Rizzolo

Versioni

Ver.	Data	Autore	Verificatore	Descrizione
0.2	3/12/2024	Diviesti Tommaso	Nome Verificatore	Continuazione e revisione del documento
0.1	29/11/2024	Lapenna Francesco	Nome Verificatore	Prima stesura del documento

Indice

1	Introduzione	3
1.1	Scopo del documento	3
1.2	Scopo del prodotto	3
1.3	Glossario	3
1.4	Riferimenti	3
1.5	Preventivo iniziale	3
2	Analisi del Capitolato	4
2.1	Obbiettivi del progetto	4
2.2	Ambito del Progetto	4
2.2.1	Inclusioni	4
2.2.2	Esclusioni	4
2.3	Funzionamento	4
2.4	Tecnologie e Strumenti Consigliati	5
2.5	Architettura proposta	5
2.6	Supporto	5
3	Pianificazione	6
3.1	Struttura del Team	6
3.1.1	Ruoli	6
3.1.2	Stakeholder	6
3.2	Budget e Risorse	6
3.2.1	TODO:	6
3.2.2	Distribuzione ore/ruolo	6
3.2.3	Distribuzione ore/membro	7
3.3	Analisi dei rischi	7
3.3.1	Definizione degli indici	7
3.3.2	Rischi	8
3.3.3	Contromisure	8
3.4	Piano di Comunicazione	9
3.5	Pianificazione delle Attività	9
3.5.1	Fasi principali	9
3.5.2	Cronoprogramma	10
4	Documentazione	11
5	Modello adottato e motivazioni	12
5.1	Sprint Log	12
5.2	RTB	12
5.2.1	Sprint 1	12
5.2.2	Sprint 2	12
5.2.3	Sprint	12
5.3	PB	12
5.3.1	Sprint 1	12
5.3.2	Sprint 2	12
5.3.3	Sprint	12

1 Introduzione

1.1 Scopo del documento

Questo documento ha lo scopo di fornire una guida dettagliata e strutturata su come il progetto verrà eseguito e gestito. In particolare, verranno trattati i seguenti argomenti:

- analisi del capitolato
- analisi delle risorse;
- analisi dei rischi;
- pianificazione;
- stime dei costi;
- modello di sviluppo adottato;
- log degli sprint;

1.2 Scopo del prodotto

Il prodotto consiste in una webapp avanzata che integra una chatbot alimentata da intelligenza artificiale, pensata per fornire informazioni precise e approfondite su una vasta selezione di bevande. L'obiettivo principale è offrire alle aziende uno strumento semplice ed efficace per accedere a dettagli fondamentali riguardo le bevande che desiderano acquistare, assicurando maggiore trasparenza e chiarezza in ogni fase del processo di selezione. Grazie alla nostra soluzione, le aziende possono ottenere risposte immediate su una serie di parametri chiave e informazioni su bibite e relativi produttori/venditori. Tutto ciò permette di ridurre incertezze e ambiguità, riducendo i rischi delle aziende legati alla scelta di prodotti non adatti alle proprie esigenze. Inoltre, le chat recenti vengono salvate e rese facilmente accessibili agli utenti, permettendo loro di rivedere in qualsiasi momento le informazioni precedentemente richieste. Questa funzionalità risulta particolarmente utile per consultare rapidamente risposte a domande frequenti o per confrontare dettagli su diverse bevande, senza dover rifare ogni ricerca, garantendo così un'esperienza più efficiente e personalizzata.

1.3 Glossario

Per avere maggiore chiarezza ed evitare ambiguità per quanto riguarda i termini utilizzati all'interno dei vari documenti, viene adottato un Glossario^G che contiene una serie di termini e relativa definizione. Grazie ad esso, sarà possibile cliccare su una determinata porzione di testo, evidenziata grazie ad uno stile specifico, all'interno di un qualsiasi documento e in questo modo si potrà visualizzare la sua definizione all'interno del Glossario^G stesso. Questa soluzione permetterà agli utenti di avere maggiore chiarezza sugli argomenti da noi trattati nei vari file di documentazione.

1.4 Riferimenti

1.5 Preventivo iniziale

Il preventivo iniziale è stato presentato durante la fase di Candidatura ed è pari a **12805€**.

Per ulteriori informazioni è possibile visualizzare il documento di analisi dei costi e assunzione impegni al seguente link:

https://code7crusaders.github.io/docs/Candidatura/Preventivo_costi.html

2 Analisi del Capitolato

2.1 Obbiettivi del progetto

- Realizzare un Assistente Virtuale che supporti i clienti nella ricerca di informazioni sui prodotti disponibili in catalogo.
- Automatizzare le risposte alle domande più frequenti, migliorando l'efficienza del servizio clienti.
- Integrare un modello LLM esistente per garantire risposte accurate e un'interfaccia user-friendly.
- Memoria a lungo termine/Salvataggio chat recenti (lo Specialist potrebbe non ricordarsi tutti i dettagli)
- Velocità di risposta e disponibilità 24/7

2.2 Ambito del Progetto

2.2.1 Inclusioni

- Database relazionale per la gestione dei dati sui prodotti.
- Integrazione di un modello LLM tramite API.
- Interfaccia utente mobile per l'interazione con l'IA.
- Funzionalità di configurazione backend per template di domande e risposte.

2.2.2 Esclusioni

- Creazione di un nuovo modello LLM.
- Supporto a lingue non previste dal modello LLM scelto.

2.3 Funzionamento

- Da un'interfaccia utente, viene catturata una domanda da parte dell'utente
- La domanda viene inoltrata al sistema attraverso delle API REST risiedenti in un Web Server
- La query ricevuta viene gestita dall'Embedding Model che trasforma la domanda in rappresentazione vettoriale
- La rappresentazione vettoriale viene utilizzata per effettuare una ricerca all'interno del database vettoriale da dove vengono reperiti i vettori più simili
- Sia la domanda sia i risultati della ricerca nel database vettoriale, vengono inviati al sistema LLM che costruirà la risposta utilizzando il contesto fornito
- Attraverso API REST, il sistema inoltra la risposta al dispositivo dell'utente

2.4 Tecnologie e Strumenti Consigliati

L'azienda proponente è disponibile a fornire i dati di un caso di studio da utilizzare per lo sviluppo del progetto. I dati potranno essere dati in ingresso al sistema così da eseguire la fase di training e poi interagire con il sistema per valutarne le prestazioni sfruttando un caso reale. Di seguito vengono suggerite alcune tecnologie utilizzabili per il sistema esposto:

- **Database:** MySQL o PostgreSQL.
- **LLM:** BLOOM o Italia by iGenius, in base alle prestazioni richieste.
- **Backend:** Node.js con Express.js o .NET.
- **Frontend:** .NET MAUI per applicazioni mobile multiplatforma.
- **API REST:** Per la comunicazione tra LLM e interfaccia utente.
- **Controllo Versione:** Git (GitHub per repository pubblico).

2.5 Architettura proposta

2.6 Supporto

Per il progetto, l'azienda proponente fornirà ampio supporto da parte del team interno in varie fasi di sviluppo. L'interazione potrà avvenire sia nei locali aziendali sia da remoto tramite chat e/o chiamate. Inoltre, mette a disposizione una serie di link e corsi utili che trattano le tecnologie relative ai sistemi LLM e allo sviluppo software.

3 Pianificazione

3.1 Struttura del Team

3.1.1 Ruoli

I ruoli in seguito descritti sono equamente divisi tra i vari componenti del Team. Ogni ruolo possiede diversi incarichi e obbiettivi:

- **Responsabile:** coordina il gruppo di lavoro, controlla le attività e gestisce le risorse. Si occupa di garantire che il progetto venga portato a termine nei tempi stabiliti e con le risorse disponibili.
- **Amministratore:** si occupa della gestione delle risorse e delle infrastrutture, incluso il setup degli strumenti di supporto alla produzione del software. Garantisce inoltre l'uso corretto delle procedure per assicurare efficienza e produttività.
- **Analista:** gioca un ruolo fondamentale nella fase iniziale del progetto. È responsabile della definizione dei requisiti e dell'analisi delle funzionalità del software, delineando i casi d'uso. Essendo necessario principalmente all'inizio del progetto, il numero di ore assegnato al ruolo è relativamente ridotto.
- **Progettista:** definisce l'architettura del software, descrivendo le componenti e le loro interazioni sulla base dei requisiti stabiliti dall'Analista. Questo ruolo ha un numero di ore significativamente elevato perché è essenziale per garantire una struttura solida, soprattutto considerando l'implementazione di modelli *LLM*, che richiedono un'architettura ben progettata e adattata a tali tecnologie.
- **Programmatore:** si occupa di scrivere il codice del software seguendo le specifiche del progettista. Il numero di ore assegnato è alto, dato che rappresenta il cuore della fase di sviluppo. Tuttavia, il ruolo ha leggermente meno ore rispetto al Verificatore, poiché abbiamo scelto di adottare una metodologia incentrata sui test, che richiede un'accurata verifica del software.
- **Verificatore:** verifica che il software e la documentazione siano conformi alle norme e alle specifiche. Questo ruolo richiede un numero di ore superiore alla media, data la necessità di test approfonditi e continui, in particolare per un progetto basato su *LLM*, dove ogni componente deve essere rigorosamente validato per garantire la precisione e l'affidabilità del sistema.

3.1.2 Stakeholder

- **Cliente:** Ergon Informatica Srl.
- **Referente interno:** Gianluca Carlesso.

3.2 Budget e Risorse

3.2.1 TODO:

- Allocazione hardware: server per database e API, risorse cloud per il modello *LLM*.
- Licenze software e costi del modello *LLM* (se applicabile).

3.2.2 Distribuzione ore/ruolo

Di seguito, si riporta il costo orario in base al ruolo assunto:

Ruolo	Costo orario (€)	per ruolo	Ore per membro
Responsabile	30	54	8
Amministratore	20	64	9
Analista	25	65	9
Progettista	25	105	15
Programmatore	15	184	26
Verificatore	15	193	28
Totale	12805	665	95

Tabella 1: Costo orario e totale

3.2.3 Distribuzione ore/membro

Tutti i componenti del Team Code7Crusaders si impegnano a dedicare un totale di **95 ore** di lavoro effettivo partizionate settimanalmente in base al ruolo di riferimento, per lo svolgimento del capitolato **C7 di Ergon Informatica**. Inoltre, ciascun membro garantisce la conclusione del progetto entro la data prevista e preventivata nel paragrafo 5 di questo documento.

Ripartizione delle ore per membro del team:

Membro	Re	Am	An	Pj	Pg	Ve	Totale
Enrico Cotti Cottini	8	9	9	15	26	28	95
Gabriele Di Pietro	8	9	9	15	26	28	95
Tommaso Diviesti	8	9	9	15	26	28	95
Francesco Lapenna	8	9	9	15	26	28	95
Matthew Pan	8	9	9	15	26	28	95
Eddy Pinarello	8	9	9	15	26	28	95
Filippo Rizzolo	8	9	9	15	26	28	95

Tabella 2: Impegni orari a persona

LEGENDA:

Re = Responsabile

Am = Amministratore

An = Analista

Pj = Progettista

Pg = Programmatore

Ve = Verificatore

3.3 Analisi dei rischi

In questa sezione vengono elencati i rischi che potrebbero verificarsi durante lo svolgimento del progetto e le relative contromisure. Ad ogni rischio è associato un **indice di Gravità e Probabilità**, in modo da poter valutare la criticità di ciascuno di essi.

3.3.1 Definizione degli indici

I fattori chiave per l'identificazione dei rischi sono:

- l'**indice**, un valore numerico incrementale che identifica univocamente il rischio
- il **tipo**, che rappresenta l'impatto che un rischio può avere sul progetto, il quale può essere basso, medio o alto

- la **gravità**, che descrive l'impatto o le conseguenze del rischio che si verificano qualora si presenti
- la **probabilità**, che rappresenta la possibilità che un rischio si verifichi

I valori dell'**Indice di Gravità** e dell'**Indice di Probabilità** sono definiti come segue:

Indice	Tipo	Gravità	Probabilità
1	Basso	Ha un impatto minimo o trascurabile sul progetto, come un lieve rallentamento che non incide sui tempi di consegna	Improbabile che si verifichi, ma esistono fattori che potrebbero contribuire alla sua realizzazione
2	Medio	Se si concretizza, richiede risorse aggiuntive o modifica parzialmente il piano di progetto, causando impatti gestibili ma che comportano sforzi supplementari	C'è una possibilità realistica che l'evento di rischio si verifichi
3	Alto	Causa ritardi significativi, aumento dei costi o degrado della qualità che incide sull'esperienza utente, richiedendo interventi importanti per mantenere il progetto nei tempi e nel budget	Esistono molti fattori o segni che indicano che il rischio potrebbe accadere, e il team considera probabile la sua manifestazione

Tabella 3: Definizione degli Indici di Gravità e Probabilità

3.3.2 Rischi

ID	Rischio	Gravità	Probabilità
1	Difficoltà nell'uso di nuove tecnologie	2	3
2	Codice non completato dal delegato	2	2
3	Riduzione del carico e delle ore di lavoro durante le festività	1	2
4	Scarsa collaborazione da parte di uno o più membri del team	3	1
5	Impegni personali e universitari	1	2
6	Deviazione dai tempi e costi previsti	3	1

Tabella 4: Analisi dei rischi

3.3.3 Contromisure

- Contromisura rischio 1:
Il gruppo si impegnerà a studiare in modo approfondite le tecnologie richieste dal capitolato in particolar modo lo studio dei *LLM*. E verranno organizzati incontri di formazione interna in modo tale da poter condividere le conoscenze acquisite per essere tutti sullo stesso livello.
- Contromisura rischio 2:
Il gruppo si impegnerà a chiedere supporto all'azienda e si cercherà di massimizzare le risorse nel team nella soluzione di un problema.
- Contromisura rischio 3:
Il gruppo cercherà di mantenere i ritmi feriali impostando un tempo minimo di lavoro settimanale.
- Contromisura rischio 4:
Il gruppo si impegna nella comprensione e nel chiarire quali siano i ruoli, inoltre una comunicazione costante e trasparente aiuterà sull'affrontare le diverse difficoltà e nel segnalare tempestivamente eventuali problemi

- Contromisura rischio 5:
Progettazione di un calendario condiviso dove ogni componente può segnalare i propri impegni personali con anticipo. Di conseguenza pianificare bene le varie attività evitando sovrapposizioni
- Contromisura rischio 6:
Monitorare il progresso delle attività e svolgere frequenti riunioni per valutare lo stato di avanzamento del progetto.

3.4 Piano di Comunicazione

- Riunioni settimanali con il team.
- Report di avanzamento per il referente aziendale ogni 2 settimane.
- Feedback continuo attraverso test intermedi.

3.5 Pianificazione delle Attività

Il gruppo Code7Crusaders si impegna a consegnare il progetto entro il 14/03/2025. La pianificazione prevede 19 settimane di lavoro, suddivise come segue:

- **Sviluppo PoC (*Proof of Concept*): 6 settimane**
- **Sviluppo MVP (*Minimum Viable Product*): 13 settimane**

3.5.1 Fasi principali

1. **Analisi dei requisiti** (2 settimane):
 - Revisione del capitolato.
 - Identificazione delle tecnologie e dei modelli LLM adatti.
2. **Progettazione** (2 settimane):
 - Progettazione architetturale.
 - Definizione dello schema del database.
3. **Sviluppo Backend** (4 settimane):
 - Configurazione del database.
 - Implementazione delle API REST.
4. **Integrazione LLM** (3 settimane):
 - Pre-processing dei dati e integrazione del modello LLM.
5. **Sviluppo Frontend** (4 settimane):
 - Creazione dell'interfaccia utente mobile.
6. **Test e validazione** (2 settimane):
 - Test funzionali e di usabilità.
7. **Rilascio e documentazione** (2 settimane).

3.5.2 Cronoprogramma



4 Documentazione

- **Da consegnare:**
 - Schema architetturale.
 - Progettazione del database.
 - Manuale per l'utilizzo della piattaforma.
 - Codice sorgente e repository Git.

5 Modello adottato e motivazioni

5.1 Sprint Log

5.2 RTB

5.2.1 Sprint 1

5.2.2 Sprint 2

5.2.3 Sprint ...

5.3 PB

5.3.1 Sprint 1

5.3.2 Sprint 2

5.3.3 Sprint ...