



UNIVERSITÀ DEGLI STUDI DI PADOVA

Corso di Laurea in Informatica
Insegnamento di Ingegneria del Software

Anno Accademico 2025/2026

Valutazione Capitolati

Sintesi comparativa e motivazione della scelta

Gruppo: Synergo Unit

Email: synergounit20@gmail.com

Versione: v1.1.1
Data: 2026-03-26

Registro Delle Modifiche

Versione	Data	Autore	Verificatore	Descrizione
1.1.1	2026-03-26	Filippo Idiometri	Sofia De Blasi	Modifica email.
1.1.0	2026-03-26	Filippo Idiometri	Sofia De Blasi	Aggiornamento pedici.
1.0.0	2026-03-24	Responsabile: Sofia De Blasi		Approvazione per ingresso in baseline.
0.1.1	2026-03-24	Filippo Idiometri	Armando Moda Scarati	Correzione ortografica.
0.1.0	2026-03-24	Filippo Idiometri	Armando Moda Scarati	Modifica criteri di valutazione.
0.0.0	2026-03-17	Anton Ricardo Rupi	Sofia De Blasi	Prima stesura del documento.

Indice

Registro Delle Modifiche	1
1 Introduzione	3
1.1 Scopo del Documento	3
1.2 Criteri di Valutazione	3
2 Confronto Capitolati	4
3 Analisi Dettagliata per Capitolato	4
3.1 Capitolato C1: Automated EN18031 Compliance Verification	4
3.2 Capitolato C2: Code Guardian	5
3.3 Capitolato C4: L'app che Protegge e Trasforma	6
3.4 Capitolato C5: NEXUM	7
3.5 Capitolato C6: Second Brain	9
4 Capitolato Selezionato: Second Brain	11
4.1 Motivazioni della Scelta	11
4.2 Rischi Principali e Mitigazioni	11
5 Conclusioni	11

1 Introduzione

1.1 Scopo del Documento

La stesura del presente documento ha lo scopo di presentare l'analisi dei capitolati esaminati, al fine di individuare quello più adatto alle nostre competenze e ai nostri obiettivi.

1.2 Criteri di Valutazione

Per garantire una scelta oggettiva, il gruppo ha definito i seguenti criteri di valutazione. A ciascun criterio è stato assegnato un punteggio da 1 (pessimo) a 5 (ottimo):

- **V1 - Interesse tecnico:** Livello di interesse nei confronti del capitolato presentato
- **V2 - Adeguatezza della complessità:** Quanto è adeguato il livello di complessità del progetto rispetto alle limitazioni presenti
- **V3 - Chiarezza requisiti:** Quanto chiaramente i requisiti di progetto sono stati esposti dall'azienda
- **V4 - Qualità rapporto proponente:** Livello di qualità del rapporto con l'azienda
- **V5 - Fattibilità tecnica:** Fattibilità del progetto a livello tecnico
- **V6 - Fattibilità economica:** Fattibilità del progetto a livello economico
- **V7 - Allineamento competenze:** Allineamento delle competenze del gruppo con quelle richieste dal capitolato

2 Confronto Capitolati

La seguente tabella riassume i punteggi (da 1 a 5) assegnati dal gruppo a ciascun capitolato analizzato, basandosi sulla presentazione iniziale, sullo studio dei documenti forniti, sugli scambi email con i proponenti e sull'analisi interna.

Capitolato	V1	V2	V3	V4	V5	V6	V7	Totale
C1: Automated EN18031 Compliance Verification	2	3	2	3	2	3	2	17
C2: Code Guardian	3	3	4	3	3	4	3	23
C4: L'app che Protegge e Trasforma	2	1	3	3	2	2	2	15
C5: NEXUM	3	3	5	3	2	2	2	20
C6: Second Brain	3	5	4	5	5	5	5	32

**Legenda Criteri: V1=Interesse, V2=Complessità, V3=Chiarezza req., V4=Rapporto prop., V5=Fatt. tecnica, V6=Fatt. economica, V7=Allineamento.*

3 Analisi Dettagliata per Capitolato

3.1 Capitolato C1: Automated EN18031 Compliance Verification

Dominio applicativo:

Sistemi software per la verifica automatizzata della conformità normativa e della sicurezza informatica di dispositivi radio e IoT.

Requisiti principali:

- Importazione di documenti in formati standard (es. CSV, XML, JSON).
- Esecuzione automatizzata dei decision tree_G per ciascun requisito EN18031.
- Rispetto delle dipendenze gerarchiche tra requisiti.
- Output chiaro (Not applicable / Pass / Fail) per ciascun requisito.
- Dashboard di visualizzazione con accesso ai risultati aggregati.
- Traduzione dei decision tree da una rappresentazione testuale o grafica in un formato facilmente modificabile con file esterni che ne descrivono i percorsi e i nodi (es. XML, JSON).

Tecnologie:

- Si predilige l'uso di Python 3.x, gestito con un Python Packaging (es. file pyproject.toml), per la parte di Backend.

Pro e contro:**• Pro:**

- contesto reale e attuale visto che affronta una necessità concreta legata all'entrata in vigore della norma EN18031,
- tecnologie aperte in quanto l'unico vincolo è dato dall'uso di python nel backend,
- approccio Agile.

• Contro:

- ambito complesso visto che si parla di norme legislative,
- dipendenza da documenti esterni come la descrizione delle componenti di rete.

Valutazione sintetica:

Il progetto risponde a un'esigenza industriale attuale e presenta un buon livello di definizione con requisiti chiari, nonostante ciò il suo essere strettamente legato all'ambito legislativo ha diminuito la volontà del gruppo a perseguirlo.

3.2 Capitolato C2: Code Guardian

Dominio applicativo:

piattaforme software per l'analisi automatizzata, il monitoraggio della qualità e la sicurezza del codice sorgente, basate su architetture multi-agente.

Requisiti principali:

- Sistema ad agenti con orchestratore: architettura a più agenti coordinati da un orchestratore centrale.
- Analisi repository_G GitHub: identificare linguaggio, framework, librerie e versioni
- Copertura test: valutare presenza e qualità dei test unitari rispetto a standard minimi.
- Analisi sicurezza OWASP_G: audit completo tramite strumenti e regole OWASP.
- Analisi documentazione: verifica della presenza e qualità di README, API_G docs e guide.
- Report dashboard web: interfaccia React che mostri lo stato del progetto (aggiornato, documentato, sicuro, con test).
- Proposte di remediation: segnalazioni con suggerimenti concreti su come migliorare codice, sicurezza e test.
- Demo finale funzionante + documentazione tecnica e di progetto.

Tecnologie:

- Backend / orchestratore: Node.js, Python
- Frontend: React.js

- Database: MongoDB_G o PostgreSQL_G
- CI/CD_G e integrazioni: GitHub Actions
- Architettura cloud: AWS_G

Pro e contro:

- **Pro:**
 - l'uso di sistemi multi-agente e LLM_G è un ambito all'avanguardia e molto richiesto dal mercato,
 - VarGroup propone di offrire un supporto costante del progetto sia sul design thinking che sulle tecnologie.
- **Contro:**
 - l'orchestrazione di agenti autonomi è complessa e il rischio che gli agenti forniscano "allucinazioni" è alto,
 - generare codice per correggere vulnerabilità senza alterare il funzionamento originale del software target è un task_G estremamente delicato,
 - raggiungere il 70% di code coverage_G su un'architettura distribuita e basata su agenti IA richiede uno sforzo notevole.

Valutazione sintetica:

Il progetto è in linea con le tendenze attuali di automazione e architetture ad agenti. Tuttavia, l'orchestrazione di un sistema a più agenti, risulta essere un obiettivo complesso e che potrebbe portare ad un incremento significativo di tempo per la realizzazione del progetto.

3.3 Capitolato C4: L'app che Protegge e Trasforma

Dominio applicativo:

applicazioni mobili intelligenti per la prevenzione della violenza di genere e il supporto alla sicurezza e al benessere delle persone.

Requisiti principali:

- Rilevamento e Alert
- Risorse e Supporto
- Funzionalità di Sicurezza Personalizzate
- Formazione e Prevenzione
- Community di Supporto

Tecnologie:

- AWS Lambda - Microservizi

- Database (DynamoDB, RDS)
- Amazon S3
- Code di Messaggi (Kinesis/SQS)
- Lambda - Elaborazione Asincrona
- Amazon SageMaker e Amazon Bedrock
- Amazon Cognito
- AWS Step Functions
- CloudWatch

Pro e contro:

- **Pro:**
 - vengono esplicitati requisiti di sicurezza e privacy che sono fondamentali per un'app che gestisce dati sensibili,
 - l'architettura che viene proposta è moderna ed in linea con le necessità del mercato,
 - l'azienda mette a disposizione referenti per il supporto tecnico specifici rispetto alle varie tecnologie.
- **Contro:**
 - l'ambito del progetto è estremamente ampio e complesso,
 - l'implementazione dei requisiti di sicurezza è complessa e stringente parlandosi di dati sensibili,
 - l'utilizzo di modelli di IA soprattutto negli ambiti proposti dal progetto ("Detective delle relazioni" e "Specchio intelligente") potrebbe rivelarsi complesso data la specificità e accuratezza di cui necessitano.

Valutazione sintetica:

Il progetto nonostante il supporto proposto dall'azienda è comunque molto complesso ed ampio il che potrebbe rivelarsi problematico dati i tempi stretti di cui il gruppo dispone.

3.4 Capitolato C5: NEXUM

Dominio applicativo:

Il progetto mira a potenziare la piattaforma NEXUM, già esistente ed utilizzata da EGGON, con due nuove funzionalità:

- AI Assistant Generativo ovvero uno strumento sulla dashboard amministrativa per aiutare HR e amministratori a creare comunicazioni aziendali (composte da testo, titolo, immagine) partendo da un prompt_G e scegliendo un tono ed uno stile;

- AI Co-pilot per Consulenti del Lavoro (CdL) cioè un modulo per la gestione documentale che permette l'automatizzazione (grazie al riconoscimento OCR_G e classificazione AI) di riconoscimento del documento, estrazione dei destinatari, generazione lista di distribuzione, invio tracciato e conservazione dei documenti.

Requisiti principali:

- I nuovi moduli dovranno essere integrabili con l'infrastruttura esistente di NEXUM,
- Ai Assistant deve permettere la creazione di contenuti completi (titolo, testo, immagine di copertina) a partire da un prompt con adattamento automatico del tono e dello stile rispetto a quelli selezionati, salvataggio del prompt, sistema di rating e possibilità di revisione_G.

Tecnologie:

- Frontend(Angular, Next.js)
- Backend(Ruby on Rails,AWS ECS, Sidekiq)
- Database e Storage(Amazon RDS for PostgreSQL per database,Elasticache for Redis per cache/sessioni,AWS S3 per storage documenti)
- Servizi AWS Core(Amazon Cognito per l'autenticazione,Amazon SES per email,AWS WAF, Shield, KMS, Secrets Manager per la sicurezza,CloudWatch, X-Ray per il monitoraggio)

Pro e contro:

- **Pro:**
 - i requisiti funzionali sono ben dettagliati,
 - lo schema architetturale proposto è in linea con il mercato del lavoro,
 - la presenza della tabella con rischi, impatti e mitigazioni è molto utile per avere già un'idea dei problemi che potranno presentarsi e di come affrontarli.
- **Contro:**
 - i requisiti del progetto sono molti e sono tutti obbligatori,
 - i criteri di accettazione sono molto stringenti ed in particolare essendo legati all'IA questi potrebbero essere difficili da raggiungere indipendentemente dal lavoro del gruppo,
 - l'integrazione del codice con un core già esistente potrebbe risultare complesso sia da un punto di vista di apprendimento che di scrittura del codice conforme a quello già presente.

Valutazione sintetica:

NEXUM è un progetto che permette l'apprendimento di tecnologie rilevanti al mondo del lavoro, inoltre l'interazione con un core già presente sarà ulteriormente utile all'apprendimento. Il problema più grande del progetto è la sua ampiezza data dal fatto che tutti i requisiti proposti sono obbligatori e dall'ampio uso di tecnologie nuove per il gruppo.

3.5 Capitolato C6: Second Brain

Dominio applicativo:

Il progetto “Second Brain”, dell’azienda Zucchetti, propone la realizzazione di un’applicazione che permetta la verifica della potenzialità degli LLM nell’aiutare una persona nei compiti legati alla creazione di testi, allo sviluppo di idee e brainstorming. La realizzazione concreta del progetto consiste in un editor di testo online, con la semplicità come obiettivo primario, basato sul linguaggio Markdown. Protagonista dell’applicazione sarà un LLM: questo permetterà infatti, tramite delle istruzioni scritte in linguaggio naturale, di migliorare, correggere, tradurre o riscrivere da zero il testo su cui l’utente sta lavorando.

Requisiti principali:

- creazione di un editor di testo Markdown basato su pagina web
- creazione di una sezione di presentazione grafica applicando lo stile Markdown sotteso dai marcatori presenti nel testo
- possibilità di modifica layout (solo editor di testo, solo presentazione grafica, entrambe)
- accesso ad un LLM in modo che possa operare sull’intero testo o parti di testo
- impartire comandi base al LLM per:
 - Riassunto del testo.
 - correzione grammaticale.
 - riscrittura con modifica
 - riscrittura con traduzione in altra lingua
 - riscrittura da zero del testo (tramite prompt specifico)
- Implementazione di comandi di critica del testo secondo il modello dei “sei cappelli per pensare” di Edward De Bono.
- salvataggio e lettura delle note Markdown in un file di testo (tramite download).
- OPZIONALE: gestire il salvataggio delle note di testo in un database remoto, salvataggio di nuovo testo, recupero e modifica di testo salvato in precedenza, possibilità di associazione di più file di testo tramite link Markdown, utilizzo dei testi collegati in fase di creazione/modifica da parte del LLM.

Tecnologie:

- HTML e CSS (pagina web con edit-area)
- Markdown (linguaggio di markup per l’editor)
- API LLM compatibili con le specifiche OpenAI

Pro e contro:

- **Pro:**
 - i vincoli del progetto sono in linea con i tempi a disposizione del gruppo,

– le tecnologie richieste sono perlopiù già conosciute dal gruppo.

• **Contro:**

– la completa riuscita del progetto si basa sull'interazione con un LLM e quindi dalla qualità dei prompt che vengono ad esso forniti in particolare per l'implementazione della critica secondo il modello di pensiero dei sei cappelli per pensare

Valutazione sintetica:

Il progetto sembra essere il più fattibile tra quelli proposti sia da un punto di vista di requisiti minimi che per le richieste tecnologiche. Il gruppo ha già una conoscenza di parte delle tecnologie richieste il che permette di non dover partire da zero, la novità starà nella capacità di saper implementare ed utilizzare un modello IA per delle richieste specifiche.

4 Capitolato Selezionato: Second Brain

4.1 Motivazioni della Scelta

Il gruppo ha scelto questo progetto in quanto esso sembra essere il più fattibile da realizzare tra quelli proposti dati i tempi limitati. Il progetto inoltre, dato che una parte delle tecnologie è già in parte conosciuta dal gruppo, permetterà ad esso di concentrarsi sull'utilizzo ed interazione con un LLM in linea con le tendenze del mondo del lavoro attuale.

4.2 Rischi Principali e Mitigazioni

Rischio	Strategia di Mitigazione
Qualità e stabilità dei risultati generati dall'LLM	Sviluppo di prompt in modo iterativo
Latenza e usabilità delle operazioni LLM	Possibilità di scelta tra diversi modelli

5 Conclusioni

Il gruppo ha scelto il capitolato Second Brain in quanto unisce il prospetto di un lavoro utile ed interessante insieme a quello di un progetto della giusta complessità per lo sviluppo dati i limiti temporali presenti.