



Università degli Studi di Padova  
Laurea: Informatica  
Corso: Ingegneria del Software  
Anno Accademico: 2025/2026



**NullPointers**  
Group

Gruppo: NullPointers Group  
Email: nullpointersg@gmail.com

# Valutazione dei Capitolati d'Appalto

Stato	Approvato
Versione	1.0
Data ultima modifica	30/10/2025
Destinatari	Prof. Tullio Vardanega Prof. Riccardo Cardin NullPointers Group

## Registro delle modifiche

Vers	Data	Autore	Ruolo	Descrizione
1.0	30 - 10	Laura Pieripolli Matteo Mazzaretto	-	Revisione e Approvazione del documento
0.2	30 - 10	Lisa Casagrande Luca Marcuzzo	-	Elaborazione del documento
0.1	27 - 10	Lisa Casagrande	-	Creazione e stesura documento

# Contents

<b>1</b>	<b>Introduzione</b>	<b>4</b>
<b>2</b>	<b>Capitolato Selezionato: C8 - SmartOrder</b>	<b>4</b>
2.1	Descrizione . . . . .	4
2.2	Dominio . . . . .	4
2.3	Motivazioni della scelta . . . . .	4
2.4	Conclusione . . . . .	5
<b>3</b>	<b>Capitolati di Interesse</b>	<b>5</b>
3.1	Capitolato C2 – CodeGuardian . . . . .	5
3.1.1	Descrizione . . . . .	5
3.1.2	Dominio . . . . .	5
3.1.3	Punti di forza . . . . .	6
3.1.4	Criticità riscontrate . . . . .	6
3.1.5	Conclusioni . . . . .	6
3.2	Capitolato C9 – View4Life . . . . .	6
3.2.1	Descrizione . . . . .	6
3.2.2	Dominio . . . . .	7
3.2.3	Punti di forza . . . . .	7
3.2.4	Criticità riscontrate . . . . .	7
3.2.5	Conclusioni . . . . .	8
<b>4</b>	<b>Altri Capitolati Valutati</b>	<b>8</b>
4.1	Capitolato C1 – Automated EN18031 Compliance Verification . . . . .	8
4.1.1	Descrizione . . . . .	8
4.1.2	Punti di forza . . . . .	8
4.1.3	Criticità riscontrate . . . . .	8
4.2	Capitolato C3 – DIPReader . . . . .	9
4.2.1	Descrizione . . . . .	9
4.2.2	Punti di forza . . . . .	9
4.2.3	Criticità riscontrate . . . . .	9
4.3	Capitolato C4 – L'app che protegge e trasforma . . . . .	9
4.3.1	Descrizione . . . . .	9
4.3.2	Punti di forza . . . . .	10
4.3.3	Criticità riscontrate . . . . .	10
4.4	Capitolato C5 – NEXUM . . . . .	10
4.4.1	Descrizione . . . . .	10
4.4.2	Punti di forza . . . . .	10
4.4.3	Criticità riscontrate . . . . .	11
4.5	Capitolato C6 – Second Brain . . . . .	11
4.5.1	Descrizione . . . . .	11
4.5.2	Punti di forza . . . . .	11
4.5.3	Criticità riscontrate . . . . .	11
4.6	Capitolato C7 – Sistema di acquisizione dati da sensori . . . . .	12
4.6.1	Descrizione . . . . .	12
4.6.2	Punti di forza . . . . .	12
4.6.3	Criticità riscontrate . . . . .	12

# 1 Introduzione

Nel presente documento viene esposta l'analisi dei capitolati d'appalto che abbiamo esaminato al fine di individuare quello più adatto alle competenze e agli obiettivi del team **"NullPointers Group"**.

L'elaborato illustra in modo chiaro le motivazioni che ci hanno portato a scegliere di candidarci per il capitolato **C8 - SmartOrder**, proposto da Ergon, evidenziandone gli aspetti che abbiamo ritenuto più stimolanti e coerenti con il nostro percorso di crescita.

Sono inoltre analizzate le altre due proposte che abbiamo preso in forte considerazione durante la fase di valutazione: il capitolato **C2 - CodeGuardian** proposto da Var Group e il capitolato **C9 - View4Life** la cui azienda proponente è Vimar. Per ciascuno di essi vengono esposti i principali punti di forza e di debolezza, così da argomentare in modo oggettivo la decisione finale di privilegiare **SmartOrder** come progetto di riferimento per il gruppo.

## 2 Capitolato Selezionato: C8 - SmartOrder

### 2.1 Descrizione

**Proponente:** Ergon Informatica Srl

Il capitolato "SmartOrder" proposto da Ergon Informatica Srl, richiede lo sviluppo di una piattaforma intelligente in grado di recepire ordini cliente da input multimodali non strutturati - come email, messaggi vocali o immagini - e trasformarli automaticamente in ordini strutturati e pronti per l'integrazione nei sistemi gestionali (ERP) aziendali.

### 2.2 Dominio

**Dominio tecnologico:** Il progetto ci permetterà di immergervi in un ecosistema tecnologico all'avanguardia, basato su intelligenza artificiale multimodale. Nello specifico, lavoreremo con:

- **Elaborazione del Linguaggio Naturale (NLP):** richiede l'utilizzo di Large Language Models (LLM) per l'interpretazione semantica del testo. Il capitolato cita esplicitamente modelli come BERT, RoBERTa e GPT;
- **Computer Vision e OCR:** È richiesto un modulo per l'elaborazione di immagini, che deve integrare motori OCR (come Tesseract o EasyOCR) per l'estrazione del testo e modelli di visione artificiale (ad esempio CNN o Vision Transformer - ViT) per l'analisi del contenuto visivo;
- **Speech-to-Text:** ovvero un sistema di trascrizione audio per gestire gli input vocali. Il capitolato menziona esplicitamente Whisper come tecnologia di riferimento per questa funzionalità;
- **Architettura Software Modulare a Layer:** Il capitolato impone un'architettura di sistema scomposta in layer ben definiti (Raccolta Input, Pre-processing, Estrazione Feature, Fusione Multimodale, Interpretazione Semantica, Validazione, Output).

**Dominio applicativo:** Il progetto si inserisce nel contesto della digitalizzazione dei processi aziendali e dell'automazione della gestione degli ordini. La piattaforma è destinata ad aziende che ricevono ordini attraverso molteplici canali (email, chat, telefono, immagini) e necessitano di standardizzare e automatizzare il processo di inserimento nei sistemi gestionali.

### 2.3 Motivazioni della scelta

La decisione di candidarci per il capitolato SmartOrder è maturata dopo un'attenta analisi che ha evidenziato diversi elementi distintivi e fortemente appealing:

- **Tecnologie Moderne:** L'opportunità di lavorare con LLM, pipeline multimodali e modelli all'avanguardia rappresenta un'occasione unica per acquisire competenze estremamente ricercate nel mercato del lavoro, specialmente nel campo dell'AI applicata.

- **Problema reale con impatto aziendale:** Sviluppo di una soluzione concreta per un’azienda ERP. Il problema della gestione di dati non strutturati (email, audio, immagini) è reale e molto sentito nel mondo business, significa creare qualcosa di tangibile e utile.
- **Architettura Chiara e Modulare:** L’architettura a layer (raccolta, pre-processing, fusione, interpretazione, validazione) è chiara e ben definita, la sua modularità ci permetterà di suddividere il lavoro in modo efficace, favorendo un coordinamento ottimale all’interno del team. Oltre tutto ci interessa l’idea di creare architetture modulari che possano crescere nel tempo applicando e comprendendo a fondo i principi di ingegneria del software necessari per costruire sistemi robusti e manutenibili.
- **Flessibilità Tecnologica:** Il capitolo suggerisce tecnologie, ma non impone stack rigidi. Si può scegliere tra React, Angular, Blazor, diversi LLM, database, ecc. Libertà di adottare le tecnologie con cui il gruppo ha più familiarità o che preferisce imparare.
- **Supporto Aziendale e Dati Reali:** La disponibilità di Ergon a fornire un referente interno e dati reali per il testing è un valore aggiunto inestimabile. Ci permetterà di confrontarci con le complessità di un progetto concreto e di validare le nostre soluzioni su casi d’uso realistici.
- **Apprendimento di architetture scalabili:** Ci interessa l’idea di creare architetture modulari che possano crescere nel tempo, inoltre riteniamo che ci possa consentire di applicare e comprendere a fondo i principi di ingegneria del software necessari per costruire sistemi robusti e manutenibili.

## 2.4 Conclusione

La nostra scelta è ricaduta su SmartOrder perché crediamo che colga perfettamente l’obiettivo di un progetto di ingegneria del software: affrontare una sfida reale con tecnologie attuali. Questo capitolo ci dà l’opportunità di lavorare su un problema concreto – l’elaborazione di ordini multimodali – utilizzando strumenti all’avanguardia come l’AI e i modelli di linguaggio, che sono sempre più richiesti nel settore. Rispetto ad altre proposte, abbiamo apprezzato particolarmente l’equilibrio tra innovazione e struttura: l’architettura a strati è ben definita e ci guida nello sviluppo, ma lascia spazio alle nostre scelte progettuali. Questo ci permetterà di applicare le conoscenze acquisite durante il corso, sperimentando allo stesso tempo con tecnologie che sono al centro del panorama tech odierno. Vediamo in SmartOrder non solo un progetto universitario, ma un’occasione significativa per costruire competenze che saranno preziose per la nostra carriera futura.

## 3 Capitolati di Interesse

### 3.1 Capitolato C2 – CodeGuardian

#### 3.1.1 Descrizione

**Proponente:** Var Group S.p.A.

Il capitolo CodeGuardian prevede la realizzazione di una piattaforma web basata su un sistema ad agenti capace di analizzare repository software per valutarne la qualità del codice, la sicurezza e la manutenzione. Il sistema sarà in grado di fornire report automatici riguardanti test, sicurezza e documentazione, oltre a proporre suggerimenti (remediation) per correggere eventuali vulnerabilità, test mancanti o scarsa qualità del codice.

#### 3.1.2 Dominio

Il **dominio tecnologico** si fonda su strumenti moderni e ampiamente utilizzati:

- Node.js e Python per lo sviluppo backend e dell’orchestratore;
- React.js per il frontend e la creazione della dashboard;
- MongoDB o PostgreSQL per la gestione dei dati;
- GitHub Actions per integrazione CI/CD;
- AWS come infrastruttura cloud.

L'approccio modulare è un punto di forza, con la possibilità di estendere la piattaforma mediante nuovi agenti o funzionalità.

**Dominio applicativo:** Il progetto si inserisce nel contesto della qualità del software e DevSecOps. La piattaforma è destinata a team di sviluppo, project manager e security officer che necessitano di strumenti automatizzati per monitorare e migliorare continuamente la qualità del codice, la sicurezza e la documentazione dei propri progetti software.

### 3.1.3 Punti di forza

- **Attualità:** Il progetto è molto attuale e rilevante, poiché affronta il tema della qualità e sicurezza del codice, aspetti fondamentali nello sviluppo moderno.
- **Tecnologie moderne:** Permette di lavorare con tecnologie moderne e richieste dal mercato del lavoro come React, Node.js, Python e AWS, permettendoci di acquisire competenze spendibili in diversi contesti.
- **Approccio multi-agente:** L'approccio multi-agente favorisce la comprensione di architetture distribuite e modulari.
- **Supporto:** Il supporto di Var Group è strutturato e continuo, l'azienda infatti prevede sessioni di mentoring, incontri tecnici e revisione del codice, offrendo un ambiente di apprendimento professionale.

### 3.1.4 Criticità riscontrate

Nonostante il grande appeal, abbiamo identificato alcune sfide che richiedono una attenta valutazione:

- **Complessità dell'architettura ad agenti:** La progettazione e implementazione di un sistema multi-agente coordinato rappresenta una sfida tecnica significativa, che richiede competenze specializzate e un notevole impegno progettuale.
- **Costruzione agenti:** Il progetto richiede una costruzione con un livello di dettaglio avanzato per ogni agente. Oltre tutto ogni agente deve avere la possibilità di evolvere.
- **Mancanza di focus specifico:** L'assenza di un focus applicativo ristretto (ad esempio, solo sicurezza o solo analisi del codice) rischia di rendere il progetto troppo vasto, con il pericolo di sviluppare molte funzionalità in modo superficiale piuttosto che poche in modo solido.

### 3.1.5 Conclusioni

Il capitolato C2 propone un progetto di indubbio appeal tecnologico, perfettamente in linea con le tendenze emergenti. Tuttavia, dopo un'attenta analisi, ci siamo resi conto che realizzare questo sistema sarebbe stato più complesso del previsto, dover gestire tanti agenti diversi, analizzare così tanti aspetti del codice e rispettare tutti i requisiti richiesti ci sembrava troppo impegnativo per il tempo a disposizione. Pertanto queste considerazioni ci hanno portato a orientare la nostra preferenza verso un capitolato che, pur mantenendo un alto profilo tecnico, presenta un perimetro e una complessità più gestibili per le nostre attuali competenze e per la tempistica del progetto formativo. Nonostante le sfide tecniche identificate, il gruppo riconosce il potenziale del capitolato. L'idea di realizzare una piattaforma di analisi e remediation automatizzata rimane molto interessante.

## 3.2 Capitolo C9 – View4Life

### 3.2.1 Descrizione

**Proponente:** Vimar S.p.A.

Il capitolato C9, proposto da Vimar S.p.A., richiede la progettazione e lo sviluppo di View4Life, una piattaforma cloud completa con applicativo web responsive per la gestione centralizzata degli impianti domotici

Vimar all'interno di residenze protette per anziani. La piattaforma deve interfacciarsi con i dispositivi domotici Vimar (attuatori, sensori di presenza UWB, termostati) tramite le API pubbliche dello standard KNX IoT, fornendo al personale sanitario strumenti per il monitoraggio della sicurezza, la gestione degli allarmi e l'analisi dei consumi energetici.

### 3.2.2 Dominio

**Dominio tecnologico:** Il proponente impone alcuni vincoli tecnologici precisi:

- KNX IoT 3rd-party API: Obbligatorio per l'integrazione con gli impianti domotici, incluso l'uso di OAuth2 e del meccanismo di notifiche push (subscription);
- Docker e Docker-compose: Obbligatori per l'infrastruttura cloud, per garantire portabilità e riproducibilità (Infrastructure as Code);
- GIT e Licenza Open Source: Il codice deve essere versionato e pubblicamente accessibile con licenza open source (es. MIT, Apache 2);
- Tecnologie suggerite: Per il frontend (Angular, React), il backend (Node.js, Java Spring, Python Flask/FastAPI) e il cloud (AWS LightSail/EC2, servizi serverless per notifiche).

**Dominio applicativo:** Il progetto si inserisce nel contesto della domotica residenziale applicata al sociale e all'assistenza. Il prodotto sarebbe destinato al personale sanitario di strutture per anziani, con l'obiettivo di migliorare la sicurezza degli ospiti (rilevamento cadute, allarmi, ...) e l'efficienza gestionale della struttura attraverso il monitoraggio energetico e degli impianti.

### 3.2.3 Punti di forza

- **Tecnologie allineate al mercato del lavoro:** Il progetto offre l'opportunità di lavorare con uno standard industriale emergente (KNX IoT) combinato a pratiche di sviluppo moderne (Infrastructure as Code, containerizzazione).
- **Progetto dal reale impatto sociale:** Sviluppare una piattaforma per migliorare la sicurezza e la qualità della vita degli anziani conferisce al lavoro un significato etico importante.
- **Kit hardware fornito:** Vimar mette a disposizione un kit fisico completo di dispositivi per testing reale.
- **Spazio alla progettazione:** Vimar lascia libertà di scelta sullo stack tecnologico e possibilità di realizzare una libreria SDK riutilizzabile.

### 3.2.4 Criticità riscontrate

- **Complessità integrazione KNX IoT:** La tecnologia KNX IoT e il flusso OAuth2 rappresentano una curva di apprendimento ripida, essendo tecnologie relativamente nuove e complesse.
- **Vincoli stringenti sui test:** La richiesta di code coverage del 75% per test di unità/integrazione e 80% per test end-to-end è molto ambiziosa e rischia di richiedere troppo tempo.
- **Dipendenza da hardware fisico:** La necessità di interfacciarsi con dispositivi reali introduce variabili non controllabili e potenziali complicazioni nel testing.
- **Comunicazione con proponente:** Gli incontri bisettimanali (SAL) sono un pro, ma la richiesta di preavviso di 1-2 settimane per incontri tecnici extra è un enorme contro. Se un blocco tecnico critico emerge due giorni dopo un SAL, il team potrebbe rimanere bloccato per 10-12 giorni, un lusso che un progetto universitario con scadenze fisse non può permettersi.

### 3.2.5 Conclusioni

Il capitolato View4Life rappresenta un'opportunità tecnologica di alto profilo, che combina domotica, cloud computing e standard industriali emergenti. Tuttavia, la complessità tecnica (KNX IoT, OAuth2, Infrastructure as Code) unita ai vincoli stringenti (test coverage, dipendenza da hardware) lo rendono una sfida particolarmente impegnativa. Il progetto richiederebbe un investimento significativo in termini di apprendimento e sviluppo, con rischi di ritardi legati all'integrazione con componenti fisici e alla curva di apprendimento delle tecnologie obbligatorie, senza contare le possibili attese legate alla gestione dei meeting, che richiedono un preavviso di 1-2 settimane per essere schedulati. Nonostante il grande potenziale formativo e l'attrattiva del dominio applicativo, queste considerazioni ci portano a valutare che il progetto potrebbe risultare troppo ambizioso per le tempistiche e le risorse disponibili.

## 4 Altri Capitolati Valutati

### 4.1 Capitolato C1 – Automated EN18031 Compliance Verification

#### 4.1.1 Descrizione

**Proponente:** Bluewind S.r.l.

Il capitolato propone lo sviluppo di un software in grado di automatizzare la verifica della conformità alla norma EN18031, relativa ai requisiti di sicurezza informatica per dispositivi radio (RED 2014/53/UE). Il progetto prevede la realizzazione di un'applicazione — desktop o web — dotata di interfaccia grafica per l'esecuzione e la visualizzazione dei Decision Tree che rappresentano i requisiti dello standard. L'applicativo dovrà importare documenti tecnici, eseguire le decision tree per ogni requisito e restituire un output (“Pass”, “Fail” o “Not Applicable”), mostrando i risultati in una dashboard interattiva.

#### 4.1.2 Punti di forza

- **Struttura chiara:** Il progetto risulta ben strutturato e chiaro, con obiettivi realistici e requisiti tecnici ben definiti. Ciò permette una buona pianificazione del lavoro e una gestione efficace del gruppo.
- **Formazione specialistica:** Offre una formazione concreta nel campo della sicurezza informatica e della conformità normativa, ambito di grande attualità nel settore IT.
- **Flessibilità tecnologica:** Garantisce libertà tecnologica nella realizzazione, permettendo di scegliere tra diverse soluzioni architetturali e strumenti di sviluppo.
- **Supporto aziendale:** Il supporto aziendale di Bluewind è ben organizzato, con incontri periodici online e in presenza, garantendo un accompagnamento costante.
- **Competenze spendibili:** L'esperienza risulta altamente formativa in ottica futura, fornendo competenze spendibili nei settori dell'IoT, embedded e cybersecurity.

#### 4.1.3 Criticità riscontrate

- **Complessità concettuale:** Il progetto presenta una complessità concettuale elevata, in quanto richiede la comprensione approfondita di una norma tecnica europea e la sua traduzione in logiche software.
- **Attività prevalentemente analitiche:** Le attività sono prevalentemente analitiche e procedurali, lasciando poco spazio alla creatività o allo sviluppo di funzionalità innovative.
- **Dominio specializzato:** Il dominio applicativo è molto specifico, quindi le competenze acquisite potrebbero essere meno utili per chi desidera orientarsi verso altri ambiti, come sviluppo web o intelligenza artificiale.

## 4.2 Capitolato C3 – DIPReader

### 4.2.1 Descrizione

**Proponente:** Sanmarco Informatica S.p.a.

Il capitolato C3, proposto da Sanmarco Informatica SPA, richiede lo sviluppo di un'applicazione chiamata DIPReader. Questo strumento software ha l'obiettivo di permettere la consultazione e la ricerca offline dei documenti contenuti in un archivio digitale (Distribution Information Package - DIP) estratto da un sistema di conservazione centralizzato. L'applicazione deve essere utilizzabile direttamente dal browser o come applicazione multi-piattaforma, senza necessità di installazione.

### 4.2.2 Punti di forza

- **Tecnologie moderne e interessanti:** L'uso di un database embedded (SQLite) e, in particolare, di una libreria specializzata per l'AI come FAISS rappresenta un'opportunità di apprendere competenze tecniche all'avanguardia e molto ricercate.
- **Architettura chiara ed estendibile:** Il proponente richiede esplicitamente un'architettura che permetta di aggiungere con agilità nuove funzionalità e metodologie di reperimento dati, indicando una buona progettualità.
- **Autoconsistenza e multi-piattaforma:** Il requisito di non richiedere installazione e di funzionare su diverse piattaforma garantisce un'esperienza utente semplice e accessibile, semplificando anche la distribuzione.

### 4.2.3 Criticità riscontrate

- **Gestione di grandi volumi di dati:** La necessità di supportare ragionamenti e visualizzazioni su grandi volumi di documenti e metadati rappresenta una sfida architettonica e implementativa non banale, con potenziali criticità sulle prestazioni.
- **Complessità dell'integrazione FAISS:** L'implementazione di meccanismi di ricerca semantica tramite FAISS richiede uno studio approfondito di concetti di intelligenza artificiale (ad esempio, gli embedding) che potrebbero essere fuori dalla portata iniziale del gruppo, richiedendo un notevole investimento in tempo per l'apprendimento.
- **Obbligo di test robusti:** Il proponente richiede esplicitamente test di unità e di integrazione robusti per garantire che nuove funzionalità non impattino quelle esistenti. Questo implica un impegno aggiuntivo e una grande disciplina nel processo di sviluppo.
- **Scarso interesse del gruppo:** Nonostante la solidità del progetto, il gruppo ha valutato altri capitolati come più affini alle proprie passioni e competenze esistenti, preferendo un'applicazione con un dominio più orientato all'assistenza diretta (chatbot) piuttosto che alla gestione documentale.

## 4.3 Capitolato C4 – L'app che protegge e trasforma

### 4.3.1 Descrizione

**Proponente:** MIRIADE S.r.l.

Il capitolato C4, proposto da Miriade srl, richiede la progettazione e lo sviluppo di un'applicazione mobile innovativa finalizzata alla prevenzione e al supporto delle vittime di violenza di genere. L'app, denominata "L'app che Protegge e Trasforma", dovrà essere uno strumento attivo, intelligente e sicuro, capace di riconoscere segnali di pericolo e offrire risorse concrete per l'autonomia e la sicurezza degli utenti. Lo sviluppo dovrà avvenire per i sistemi operativi iOS e Android.

#### 4.3.2 Punti di forza

- **Altissimo impatto sociale e etico:** Il progetto ha uno scopo nobile e concreto, offrendo l'opportunità di contribuire a una causa sociale di grande rilevanza. Questo aspetto può essere una forte motivazione per il team.
- **Supporto del proponente:** Miriade si distingue per un supporto molto strutturato, offrendo referenti specifici per cloud, sviluppo, design, UX e persino per la tematica sociale della violenza di genere. Mettono a disposizione strumenti (Jira, Bitbucket), formazione iniziale sulla materia e supporto tecnico continuo.
- **Tecnologie moderne:** L'uso di un'architettura serverless su AWS, microservizi, Flutter e servizi di AI (SageMaker, Bedrock) rappresenta un'opportunità unica per apprendere un stack tecnologico molto avanzato e ricercato nel mercato.

#### 4.3.3 Criticità riscontrate

- **Complessità tecnica:** Il progetto combina una sfida tecnica notevole (architettura a microservizi, AI, sicurezza elevatissima).
- **Vincoli tecnologici impliciti:** Sebbene sia dichiarata libertà di scelta, la bozza architetturale su AWS e la forte raccomandazione per Flutter creano uno stack tecnologico di fatto preferenziale, lasciando meno spazio a soluzioni alternative rispetto ad altri capitolati.
- **Scarso interesse del gruppo:** Nonostante l'indiscutibile valore del progetto, il gruppo ha valutato altri capitolati come più affini alle proprie competenze esistenti e ai propri interessi tecnici.
- **Complessità del capitolato:** Temiamo che la complessità estrema del capitolato (microservizi, AI, sicurezza avanzata) ci porterebbe a dedicare più tempo alla configurazione infrastrutturale che ai principi solidi di ingegneria del software. Questo, unito alla mole di lavoro richiesta, metterebbe a serio rischio la nostra capacità di consegnare un prodotto completo e di qualità nei tempi previsti dal corso.

### 4.4 Capitolato C5 – NEXUM

#### 4.4.1 Descrizione

**Proponente:** Eggon S.r.l.

Il capitolato C5 riguarda lo sviluppo di nuove funzionalità AI per la piattaforma NEXUM, una soluzione digitale per la gestione HR e la comunicazione tra aziende, dipendenti e Consulenti del Lavoro. Il progetto prevede due moduli principali: un AI Assistant Generativo per la creazione automatica di contenuti aziendali e un AI Co-Pilot per automatizzare la gestione documentale degli studi di consulenza del lavoro.

#### 4.4.2 Punti di forza

- **Stack tecnologico moderno:** Utilizzo di tecnologie attuali come Ruby on Rails, Angular, AWS e integrazione di servizi AI/OCR.
- **Supporto e mentorship aziendale:** La collaborazione stretta con Eggon fornisce un supporto tecnico e metodologico che può accelerare l'apprendimento e ridurre i rischi di fallimento.
- **Dominio applicativo ben definito:** Casi d'uso chiari e circoscritti per entrambi i moduli AI
- **Opportunità di stage e collaborazioni:** Possibilità di continuare la collaborazione con l'azienda dopo il termine del progetto.

#### 4.4.3 Criticità riscontrate

- **Alta complessità di integrazione:** Necessità di interfacciarsi con un'architettura esistente e complessa (NEXUM Core).
- **Vincoli tecnologici rigidi:** Stack tecnologico largamente predefinito con poca flessibilità nella scelta degli strumenti.
- **Ambito funzionale molto ampio:** I due moduli principali contengono numerosi casi d'uso che potrebbero richiedere uno sforzo di sviluppo eccessivo.
- **Responsabilità legate all'AI in produzione:** Sviluppare funzionalità AI che andranno in produzione porta con sé la responsabilità di garantire affidabilità, assenza di bias e corretto handling degli errori, aspetti non sempre semplici da gestire.
- **Carico di lavoro non banale:** I 10 casi d'uso per il Co-Pilot + 6 per l'Assistant sono molti. Il rischio di sovraccarico o di dover ridurre l'ambito è concreto e andrebbe gestito con attenzione durante la pianificazione.

### 4.5 Capitolato C6 – Second Brain

#### 4.5.1 Descrizione

**Proponente:** Zucchetti S.p.A.

Il capitolato C6, proposto da Zucchetti S.p.A., richiede lo sviluppo di un'applicazione web per il note-taking avanzato che integri le capacità dei Large Language Models (LLM). Il sistema si basa su un editor MarkDown che permette all'utente non solo di scrivere e formattare testo, ma anche di interagire con un LLM per migliorare, analizzare e generare contenuti attraverso funzioni come riassunto, riscrittura, traduzione e critica del testo secondo il metodo dei "sei cappelli per pensare" di Edward De Bono.

#### 4.5.2 Punti di forza

- **Focus tecnologico chiaro e ben definito:** Il progetto si concentra principalmente sull'integrazione e l'ottimizzazione dei prompt per gli LLM, un ambito attuale e di grande interesse nel panorama tecnologico attuale.
- **Architettura flessibile:** La suddivisione in PoC, MVP e prodotto finale con requisiti opzionali permette una buona gestione del progetto.
- **Libertà nella scelta delle tecnologie:** Zucchetti non impone stack tecnologici specifici, lasciando al team la possibilità di scegliere gli strumenti più adatti per il frontend, backend e integrazione LLM.
- **Supporto aziendale per l'accesso agli LLM:** L'azienda mette a disposizione modelli LLM per i test e offre supporto per le parti tecniche più complesse.

#### 4.5.3 Criticità riscontrate

- **Mancanza di un dominio applicativo specifico:** Il progetto appare come uno strumento generico di note-taking senza un target di utenti ben definito o un contesto applicativo chiaro.
- **Complessità nella gestione dei prompt:** La creazione di prompt efficaci per tutte le funzionalità richieste (specialmente i "sei cappelli") potrebbe rivelarsi più complessa del previsto.
- **Alto numero di funzionalità da integrare nell'interfaccia:** Il progetto richiede l'integrazione di un ampio set di comandi e modalità di interazione (generazione, riassunto, riscrittura, traduzione, multiple forme di analisi critica del testo) all'interno di una singola interfaccia. Il rischio è di creare un'esperienza utente sovraccarica e confusa, anziché semplice e intuitiva come dichiarato negli obiettivi.

- **Strumento generico, possibile scarso appeal per il gruppo:** Il progetto mira a creare uno strumento di note-taking "generico". Se il team è composto da persone che non sono appassionate di scrittura, organizzazione della conoscenza o non utilizzano strumenti simili (come Obsidian o Notion) nella loro vita quotidiana, potrebbe risultare difficile trovare la motivazione e l'empatia necessaria per immaginare e sviluppare le funzionalità più innovative. Il rischio è di produrre un'esercitazione tecnica invece che un prodotto in cui si crede.

## 4.6 Capitolato C7 – Sistema di acquisizione dati da sensori

### 4.6.1 Descrizione

**Proponente:** M31 S.r.l.

Il progetto mira a sviluppare un sistema cloud per l'acquisizione, la gestione e la distribuzione di dati provenienti da sensori Bluetooth Low Energy (BLE) in uno scenario multi-tenant. Il sistema è articolato in tre livelli:

- Sensori BLE (esterni, non sviluppati);
- Gateway BLE-WiFi (esterni, sostituiti da un simulatore da sviluppare);
- Piattaforma Cloud (cuore del progetto, da sviluppare).

L'obiettivo è creare un'infrastruttura scalabile e sicura che gestisca l'autenticazione dei gateway, riceva dati, li renda disponibili via API (REST/GraphQL e streaming) e fornisca un'interfaccia web per la visualizzazione e l'amministrazione.

### 4.6.2 Punti di forza

- **Stack Tecnologico Moderno e Professionalizzante:** Utilizzare tecnologie richieste dal mercato come NestJS, TypeScript, Kubernetes, Docker, MongoDB, Redis e Angular potrebbero dare un vantaggio pratico notevole. È un'occasione per arricchire il curriculum con competenze concrete e spendibili.
- **Apprendimento di Architetture Complesse:** Lavorare su un'architettura a microservizi con pattern come multi-tenancy e messaggistica asincrona (NATS/Kafka) è un'esperienza formativa di alto livello. Ci permetterebbe di capire come funzionano davvero i sistemi software enterprise, andando oltre i semplici progetti monolitici universitari.
- **Simulazione di un Ambiente Lavorativo Reale:** Il capitolato è scritto in modo molto professionale, con chiari requisiti, milestone (PoC, MVP) e enfasi su testing, CI/CD e documentazione. Sarebbe un'ottima simulazione di come si lavora in un'azienda tech, preparandoci meglio per il futuro.

### 4.6.3 Criticità riscontrate

- **Complessità e Rischio di Sovraccarico:** La combinazione di microservizi, sicurezza, API differenti, UI e DevOps è molto impegnativa. Il rischio di sottostimare il tempo e le energie necessarie è alto, soprattutto se abbinato agli altri impegni universitari. Potremmo ritrovarci a dover sacrificare la qualità o a non completare tutte le funzionalità.
- **Complessità infrastrutturale:** Kubernetes, sebbene importante professionalmente, richiederebbe troppo tempo per la configurazione, distogliendo risorse dallo sviluppo delle funzionalità core del progetto.
- **Alto numero di tecnologie da padroneggiare:** Anche se lo stack è suggerito, il team dovrebbe avere familiarità pregressa o una forte capacità di apprendimento in tempi brevi su NestJS, TypeScript, un database NoSQL (MongoDB), un message broker (NATS/Kafka), Angular, e tutto l'ecosistema DevOps.
- **Preferenza per altri progetti:** Riteniamo che altri capitolati offrano un migliore bilanciamento tra complessità tecnica e apprendimento, meglio allineati con i nostri obiettivi formativi.