



## BugBusters

Email: `bugbusters.unipd@gmail.com`

Gruppo: 4

**Università degli Studi di Padova**

Laurea in Informatica

Corso: Ingegneria del Software

Anno Accademico: 2025/2026

# Resoconto Capitolati

Versione 0.3.0

<b>Redattori</b>	Alberto Autiero
<b>Verificatori</b>	[Nome Cognome]
<b>Uso</b>	Interno
<b>Destinatari</b>	Prof. Tullio Vardanega, Prof. Riccardo Cardin
<b>Data</b>	18/10/2025

### Abstract

Documento di analisi e valutazione dei capitoli proposti per l'anno accademico 2025/2026. Il documento include una valutazione dettagliata del capitolo scelto e un'analisi comparativa degli altri capitoli disponibili.

## 1 Versioni del documento

Versione	Data	Descrizione	Redatto	Verificato	Approvato
0.3.0	21/10/2025	Terminato C5, iniziato C8, C9	Linor Sadè	-	-
0.2.0	21/10/2025	Aggiunta introduzione al documento, aggiunti punti di forza e debolezza per ogni capitolato	Linor Sadè	-	-
0.1.0	20/10/2025	Prima stesura della struttura del documento	Alberto Autiero	Marco Favero	-

# Indice

<b>1</b>	<b>Versioni del documento</b>	<b>2</b>
<b>2</b>	<b>Metodologia di valutazione</b>	<b>4</b>
<b>3</b>	<b>Elenco capitoli analizzati</b>	<b>4</b>
3.1	C1 - Automated EN18031 Compliance Verification . . . . .	5
3.1.1	Breve descrizione . . . . .	5
3.1.2	Requisiti funzionali e non funzionali . . . . .	5
3.1.3	Tecnologie proposte . . . . .	5
3.1.4	Chiarimenti e colloqui con l'azienda . . . . .	5
3.1.5	Interesse del team . . . . .	5
3.1.6	Punti di forza e di debolezza . . . . .	5
3.2	C2 - Code Guardian . . . . .	6
3.2.1	Breve descrizione . . . . .	6
3.2.2	Requisiti funzionali e non funzionali . . . . .	6
3.2.3	Tecnologie proposte . . . . .	6
3.2.4	Chiarimenti e colloqui con l'azienda . . . . .	6
3.2.5	Interesse del team . . . . .	6
3.2.6	Punti di forza e di debolezza . . . . .	6
<b>4</b>	<b>C3 - DIPReader</b>	<b>7</b>
4.1	Breve descrizione . . . . .	7
4.2	Requisiti funzionali e non funzionali . . . . .	7
4.3	Tecnologie proposte . . . . .	7
4.4	Chiarimenti e colloqui con l'azienda . . . . .	7
4.5	Interesse del team . . . . .	7
4.6	Punti di forza e di debolezza . . . . .	7
<b>5</b>	<b>C4 - L'app che Protegge e Trasforma</b>	<b>8</b>
5.1	Breve descrizione . . . . .	8
5.2	Requisiti funzionali e non funzionali . . . . .	8
5.3	Tecnologie proposte . . . . .	8
5.4	Chiarimenti e colloqui con l'azienda . . . . .	8
5.5	Interesse del team . . . . .	8
5.6	Punti di forza e di debolezza . . . . .	8
5.7	C5 - Nexum . . . . .	9
5.7.1	Breve descrizione . . . . .	9
5.7.2	Requisiti funzionali e non funzionali . . . . .	9
5.7.3	Tecnologie proposte . . . . .	9
5.7.4	Chiarimenti e colloqui con l'azienda . . . . .	10
5.7.5	Interesse del team . . . . .	11
5.7.6	Punti di forza e di debolezza . . . . .	11
<b>6</b>	<b>C6 - Second Brain</b>	<b>12</b>
6.1	Breve descrizione . . . . .	12
6.2	Requisiti funzionali e non funzionali . . . . .	12
6.3	Tecnologie proposte . . . . .	12
6.4	Chiarimenti e colloqui con l'azienda . . . . .	12
6.5	Interesse del team . . . . .	12
6.6	Punti di forza e di debolezza . . . . .	12

<b>7</b>	<b>C7 - Sistema di acquisizione dati da sensori</b>	<b>13</b>
7.1	Breve descrizione . . . . .	13
7.2	Requisiti funzionali e non funzionali . . . . .	13
7.3	Tecnologie proposte . . . . .	13
7.4	Chiarimenti e colloqui con l'azienda . . . . .	13
7.5	Interesse del team . . . . .	13
7.6	Punti di forza e di debolezza . . . . .	13
<b>8</b>	<b>C8 - Smart Order</b>	<b>13</b>
8.1	Breve descrizione . . . . .	13
8.2	Requisiti funzionali e non funzionali . . . . .	14
8.3	Tecnologie proposte . . . . .	14
8.4	Chiarimenti e colloqui con l'azienda . . . . .	14
8.5	Interesse del team . . . . .	14
8.6	Punti di forza e di debolezza . . . . .	14
<b>9</b>	<b>C9 - View4Life</b>	<b>15</b>
9.1	Breve descrizione . . . . .	15
9.2	Requisiti funzionali e non funzionali . . . . .	15
9.3	Tecnologie proposte . . . . .	15
9.4	Chiarimenti e colloqui con l'azienda . . . . .	15
9.5	Interesse del team . . . . .	15
9.6	Punti di forza e di debolezza . . . . .	15

## 2 Metodologia di valutazione

Per la valutazione dei capitolati proposti, il team ha adottato una metodologia strutturata che prevede i seguenti passaggi:

1. **Discussione:** Dopo le presentazioni delle aziende, il team si è trovato “a caldo” per discutere di ciascun capitolato esaminando l’interesse verso ciascun progetto (propensione, curiosità, motivazione) e le competenze richieste (tecnologie note, tecnologie da apprendere).
2. **Analisi individuale:** Ogni membro del team ha condotto un’analisi individuale di ogni capitolato, leggendone il relativo documento di proposta, valutandone più attentamente i punti di forza e di debolezza in base a criteri predefiniti.
3. **Richiesta di chiarimenti:** In caso di dubbi o incertezze riguardo a specifici aspetti dei capitolati, il team ha preparato una lista di domande da inviare ai proponenti per ottenere chiarimenti. In seguito sono stati fissati incontri con le aziende per discutere questi punti (se necessario). In particolare ci interessava capire:
  - Il livello di supporto che l’azienda fornirà durante lo sviluppo del progetto.
  - L’importanza nella conoscenza pregressa delle tecnologie richieste.
  - ALTRO DA DEFINIRE
4. **Compilazione tabelle:** Il team si è diviso il compito di compilare le tabelle dei punti di forza e di debolezza per ogni capitolato, assicurando che ogni membro contribuisse in modo equo.

I punti 3 e 4 sono stati ripetuti più volte fino a quando il team non si è sentito sufficientemente preparato per prendere una decisione informata sul capitolato da scegliere.

Il team ha deciso di dare particolare importanza all’interesse verso l’argomento e le tecnologie rispetto alla complessità del progetto, senza tuttavia trascurare la valutazione della sua fattibilità. Crediamo che lavorare su un progetto che suscita entusiasmo e curiosità porterà a un’esperienza di apprendimento più significativa e gratificante, anche se ciò comporta affrontare sfide tecniche più impegnative.

## 3 Elenco capitolati analizzati

Di seguito sono elencati i capitolati analizzati dal team BugBusters. Per ogni capitolato, sono riportati i punti di forza e di debolezza identificati durante la fase di valutazione.

### 3.1 C1 - Automated EN18031 Compliance Verification

#### 3.1.1 Breve descrizione

#### 3.1.2 Requisiti funzionali e non funzionali

#### 3.1.3 Tecnologie proposte

#### 3.1.4 Chiarimenti e colloqui con l'azienda

#### 3.1.5 Interesse del team

#### 3.1.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• Requisiti Ben Definiti: I requisiti obbligatori e opzionali sono elencati in modo chiaro e strutturato. Questo riduce l'ambiguità e fornisce un'ottima checklist per la pianificazione e la verifica finale del progetto.</li><li>• Caso Studio Concreto: La presenza di un caso studio specifico (la macchina da caffè connessa via Wi-Fi) fornisce un dominio applicativo tangibile per testare le funzionalità, evitando di lavorare in modo troppo astratto.</li><li>• Bluewind si impegna esplicitamente per un supporto "ibrido" (online e in presenza), con incontri periodici. Questo è un enorme vantaggio, poiché fornisce un canale diretto per chiarire dubbi, ottenere feedback e allinearsi con le aspettative dell'azienda.</li></ul>	<ul style="list-style-type: none"><li>• Complessità del Dominio: Il dominio normativo (EN 18031, direttiva RED) è intrinsecamente complesso. Comprendere appieno la logica dei "Decision Tree" e le interdipendenze tra i requisiti richiederà uno sforzo iniziale significativo di analisi e studio.</li></ul>

## 3.2 C2 - Code Guardian

### 3.2.1 Breve descrizione

### 3.2.2 Requisiti funzionali e non funzionali

### 3.2.3 Tecnologie proposte

### 3.2.4 Chiarimenti e colloqui con l'azienda

### 3.2.5 Interesse del team

### 3.2.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• Tema Innovativo e ad Alto Potenziale: L'uso di un'architettura multi-agente per l'analisi automatizzata del codice è estremamente attuale e all'avanguardia.</li><li>• Dominio Concreto e Utile: La piattaforma risolve problemi reali di qualità del codice, sicurezza e manutenzione dei repository.</li><li>• Sessione di mentoring sulle tecnologie che verranno utilizzate</li></ul>	<ul style="list-style-type: none"><li>• Alta Complessità Concettuale: L'architettura multi-agente è concettualmente avanzata. Progettare un sistema dove agenti specializzati comunicano efficacemente attraverso un orchestratore richiede una solida comprensione di pattern complessi.</li><li>• Requisiti di Testing Stringenti: La richiesta del 70% di test coverage (obbligatorio) è apprezzabile professionalmente ma può essere impegnativa da raggiungere in un progetto universitario, specialmente per componenti AI.</li><li>• Non sembra esserci flessibilità nella scelta delle tecnologie da utilizzare</li></ul>

## 4 C3 - DIPReader

### 4.1 Breve descrizione

### 4.2 Requisiti funzionali e non funzionali

### 4.3 Tecnologie proposte

### 4.4 Chiarimenti e colloqui con l'azienda

### 4.5 Interesse del team

### 4.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• Dominio Specializzato e di Alto Valore: La conservazione digitale è un campo di nicchia ma cruciale, specialmente in ambito legale e amministrativo.</li><li>• Problema Concreto e Ben Definito: L'esigenza di accedere a documenti conservati digitalmente in modalità offline è un requisito reale per molti professionisti. Il progetto risolve un problema tangibile.</li><li>• Interazione con l'azienda: offre esempi di pacchetti estratti dal sistema di conservazione e la relativa documentazione</li></ul>	<ul style="list-style-type: none"><li>• Complessità del Dominio Normativo: La conservazione digitale è regolata da standard e normative complesse. Comprendere appieno il formato dei DIP e i requisiti di compliance richiederà uno sforzo iniziale significativo.</li><li>• Sfide Tecniche per l'Offline: Implementare ricerche efficienti e visualizzazioni di anteprima completamente offline, specialmente per grandi volumi di dati, presenta sfide non banali di performance e gestione della memoria.</li><li>• Ambiguità Architetturale: Non è completamente chiaro se l'applicazione debba essere una PWA, un'app desktop (Electron) o entrambe. Questa decisione avrà impatti significativi sull'architettura.</li><li>• Scope Potenzialmente Ampio: Le funzionalità opzionali come la ricerca semantica (con FAISS) e la verifica delle firme digitali sono progetti ambiziosi che potrebbero distrarre dallo sviluppo del MVP.</li><li>• Testing su Grandi Volumi: Garantire le performance con "grandi volumi" di dati potrebbe essere difficile da testare e validare in ambiente universitario.</li></ul>



## 5 C4 - L'app che Protegge e Trasforma

### 5.1 Breve descrizione

### 5.2 Requisiti funzionali e non funzionali

### 5.3 Tecnologie proposte

### 5.4 Chiarimenti e colloqui con l'azienda

### 5.5 Interesse del team

### 5.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• Impatto Sociale Elevatissimo: Il progetto ha uno scopo nobile e concretamente utile - prevenire e supportare vittime di violenza di genere. Sviluppare un'app che può potenzialmente salvare vite fornisce una motivazione etica molto forte.</li><li>• supporto Aziendale Eccezionale e Strutturato: Miriade offre un supporto completo: o Referenti specializzati per ogni area (tecnica, design, dominio sociale) o Formazione iniziale sulla tematica della violenza di genere o Strumenti professionali (Jira, Bitbucket) o Supporto multidisciplinare continuo o Possibilità di incontri in sede</li><li>• Requisiti Chiari e Dettagliati: Sia i requisiti funzionali che non funzionali sono ben specificati, con una chiara distinzione tra obbligatori e opzionali.</li><li>• Ottima possibilità riguardo il ciclo di vita dell'applicazione: analisi, progettazione, sviluppo, test di sicurezza e controllo dei contenuti etici devono essere fatti in modo chiaro e preciso, apprendendo così competenze multidisciplinari.</li></ul>	<ul style="list-style-type: none"><li>• Alta Complessità Tecnica e Progettuale: o L'architettura proposta è ambiziosa e forse eccessiva essendo per alcuni il primo affronto verso certe conoscenze o L'integrazione di AI/ML per l'analisi comportamentale richiede competenze specialistiche o L'architettura serverless su AWS con microservizi è complessa da gestire o La sicurezza dei dati è critica e richiede implementazioni robuste</li><li>• Responsabilità e Sensibilità del Dominio: L'errore in un'app di questo tipo può avere conseguenze gravi, soprattutto dal punto di vista etico. La progettazione deve essere impeccabile sotto il profilo della sicurezza e dell'affidabilità.</li><li>• Scope Molto Ampio: Le funzionalità previste sono numerose e ambiziose (rilevamento AI, allarmi silenziosi, diario criptato, moduli educativi, community). Il rischio di sovra-estendere il progetto è alto.</li></ul>

## 5.7 C5 - Nexum

### 5.7.1 Breve descrizione

NEXUM nasce come piattaforma HR evoluta, in grado di connettere aziende, collaboratori e gli studi dei CdL. Il progetto prevede lo sviluppo di nuovi moduli per la piattaforma esistente, con l'obiettivo di migliorare l'efficienza dei processi HR e offrire funzionalità innovative ai suoi utenti. Il team si occuperà di sviluppare due moduli principali:

- **Ai Assistant Generativo per HR:**

Che dovrà permettere agli utenti di creare in autonomia contenuti accattivanti con titolo, descrizione e immagine di copertina attraverso l'uso di AI generativa, adeguando tono e stile della comunicazione a quello aziendale (formale, informale ...ecc).

- **Ai Co-Pilot per i CdL:**

Deve essere in grado di riconoscere la tipologia di documenti caricati (cedolini, comunicazioni, documenti da firmare, ecc.) e i destinatari, direttamente dal documento e consegnarli ai destinatari anche in modo massivo.

### 5.7.2 Requisiti funzionali e non funzionali

Tra i requisiti funzionali del progetto vi sono:

- Integrazione con la piattaforma esistente NEXUM
- Utilizzo di tecnologie cloud (AWS)
- Implementazione delle funzionalità di AI generativa e AI Co-Pilot
- Le funzionalità devono essere disponibili sulla dashboard o sulla PWA

Tra i requisiti non funzionali vi sono:

- fluidità d'uso e interfaccia user-friendly
- le operazioni time-consuming dovranno essere delegate a sistemi batch

Altri requisiti di prestazione saranno concordati con il team in base alle tecnologie utilizzate e soluzioni proposte

### 5.7.3 Tecnologie proposte

Le tecnologie proposte per lo sviluppo del progetto includono:

- **Frontend:** Angular (dashboard amministrativa), Next.js (PWA utenti finali, hosting su AWS Amplify o S3+CloudFront)
- **Backend e API:** Ruby on Rails (stateless su ECS Fargate dietro ALB)
- **Background Jobs:** Sidekiq (su Fargate + SQS)
- **Database e cache:** PostgreSQL (RDS Multi-AZ con snapshot automatici), ElastiCache for Redis (cache e gestione sessioni)
- **Storage:** Amazon S3 (bucket separati per "uploads" e "processed", Lifecycle e legal hold opzionale)
- **Sicurezza:** KMS (gestione chiavi per S3, RDS, Secrets), Secrets Manager (credenziali, API keys, JWT secrets), Amazon Cognito (gestione identità e accessi)
- **Comunicazioni:** SES (email), SNS (notifiche push/eventi)
- **Networking:** VPC (reti private e pubbliche per ECS/RDS/Redis, con ALB e NAT Gateway), Security Groups (accesso minimo privilegio)
- **Sicurezza web:** WAF + AWS Shield (protezione ALB/CloudFront), IAM policy granulari (permessi per ECS task roles, S3, SQS, Secrets, CloudWatch)
- **Osservabilità:** CloudWatch Logs/Metrics/Alarms/X-Ray (monitoring e tracing)

- **Compliance e threat detection:** AWS Config + GuardDuty

#### 5.7.4 Chiarimenti e colloqui con l'azienda

Qui di seguito sono riportate alcune delle domande poste ad Eggon con le relative risposte fornite dall'azienda. In particolare abbiamo posto domande che riguardavano il supporto di Eggon durante lo sviluppo del progetto, l'importanza della conoscenza pregressa delle tecnologie richieste e il coinvolgimento del team nel processo SCRUM dell'azienda perché sapevamo fosse il primo anno in cui Eggon partecipava a questo tipo di progetto.

Domande	Risposte
Le tecnologie richieste per lo sviluppo del progetto sono diverse e molte sono completamente nuove per noi: quanto è rilevante per voi la conoscenza pregressa e quali sono le vostre aspettative rispetto al nostro apprendimento progressivo durante il progetto? Verrà fornito supporto o affiancamento nell'utilizzo di queste tecnologie?	<ul style="list-style-type: none"> <li>• <b>Conoscenza pregressa:</b> non vincolante. Valutiamo impegno, qualità del codice e velocità di apprendimento.</li> <li>• <b>Aspettative:</b> avanzamento sprint-by-sprint, PR piccole e frequenti, test minimi, documentazione essenziale.</li> <li>• <b>Supporto Eggon:</b> kickoff e seed repository, canale e-mail/Telegram, code review e pairing su temi critici, sandbox (API mock, S3, chiavi temporanee).</li> </ul>
Nella vostra esperienza avete già avuto modo di affidare una certa responsabilità operativa o decisionale a team che non avevano ancora esperienza nel mondo del lavoro? Se sì, quali risultati o insegnamenti ne avete tratto in termini di autonomia, qualità del lavoro e collaborazione con il vostro team interno?	<ul style="list-style-type: none"> <li>• Ci lavoriamo spesso: funziona quando suddividiamo il lavoro in milestone piccole con demo frequenti, manteniamo standard chiari (lint/test/review) e i blocchi emergono subito.</li> <li>• <b>Obiettivo:</b> autonomia crescente — più guida all'inizio, più ownership col passare degli sprint.</li> </ul>
Avete parlato di includere il team di lavoro nel vostro processo SCRUM e nelle riunioni o stand-up periodiche: quale cadenza hanno questi incontri e come si svolgono concretamente? Considerando che abbiamo anche impegni universitari, ci potete chiarire se è previsto che partecipiamo a tutte le daily stand-up o solo ad alcune delle cerimonie principali (ad esempio sprint review o retrospettive)?	<ul style="list-style-type: none"> <li>• <b>Sprint:</b> 2 settimane.</li> <li>• <b>Cerimonie:</b> <ul style="list-style-type: none"> <li>– Grooming/Planning (<math>\approx 1h</math>; nei primi sprint può servire più tempo) — obbligatoria.</li> <li>– Check-in asincroni su Telegram (daily in 3 righe: fatto / da fare / blocchi).</li> <li>– Review + Retro (<math>\approx 1h</math>) — obbligatorie con demo.</li> </ul> </li> <li>• Calendario condiviso. <ul style="list-style-type: none"> <li>– Lo costruiamo insieme attorno ai vostri impegni di studio (lezioni, esami, sessioni).</li> <li>– Una volta concordate milestone e scadenze, ci si impegna a rispettarle: fa parte del patto professionale azienda-fornitore e ci permette di coordinare bene tutto il team.</li> </ul> </li> </ul>
Quali modelli LLM specifici prevedete di utilizzare? Oppure possiamo testare con diversi provider?	<ul style="list-style-type: none"> <li>• <b>Preferenza:</b> AWS Bedrock (integrazione e governance). Tramite Bedrock possiamo usare più modelli (Claude, Llama, Mistral).</li> <li>• <b>Apertura ad alternative:</b> via libera a provider/idee creative, purché valutate su qualità output, aderenza al prompt, performance, costi e manutenibilità.</li> <li>• <b>Requisito:</b> adapter per evitare lock-in</li> </ul>

### 5.7.5 Interesse del team

Il team ha mostrato un interesse significativo verso la proposta di Eggon. Inanzitutto, i rappresentanti dell'azienda hanno subito mostrato uno stile giovanile e informale, pur mantenendo professionalità e struttura. Ciò ha avuto un impatto positivo come prima impressione.

Inoltre, il progetto proposto si allinea bene con le aspirazioni del team di acquisire esperienza pratica nello sviluppo di applicazioni reali, specialmente in un contesto aziendale. L'opportunità di lavorare su una piattaforma HR esistente come NEXUM, che ha una base di utenti reale, ha particolare attrattiva.

Infine, come già specificato all'inizio di questo documento, il team pone molta importanza al supporto che l'azienda può offrire durante lo sviluppo del progetto. Le risposte fornite da Eggon alle nostre domande hanno confermato che l'azienda è disposta a fornire un supporto strutturato e continuo, il che aumenta ulteriormente l'interesse del team verso questo capitolato.

### 5.7.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• Prodotto Reale e Integrazione con Piattaforma Esistente: NEXUM è una piattaforma HR già operativa. Sviluppare moduli che si integreranno in un prodotto commerciale fornisce un'esperienza di lavoro su codice legacy e integrazione con sistemi esistenti.</li><li>• Processo SCRUM realistico (rispettare le scadenze dell'azienda)</li><li>• Supporto dell'azienda: l'azienda ha un piano ben strutturato quali le cerimonie a cui il team deve necessariamente partecipare (con relativa durata) e il tipo di dialogo che vuole con il team (giornalmente). Questo fa capire al team che il supporto sarà concreto.</li><li>• Tecnologie definite esaustivamente e requisiti del progetto chiari, con due requisiti opzionali definiti.</li><li>• L'azienda ha fornito un documento per il capitolato completo di casi d'uso e obiettivi misurabili in milestone, rendendo la pianificazione più semplice</li><li>• Integrazione di AI. Lavorare su queste tecnologie fornisce un'esperienza preziosa vista la loro crescente importanza nel settore.</li></ul>	<ul style="list-style-type: none"><li>• NEXUM è una piattaforma già esistente e operativa, lo sviluppo di nuovi moduli (che verranno realmente utilizzati) comporta un livello di rischio elevato. L'integrazione in un prodotto reale, destinato a un mercato effettivo e a clienti concreti, implica vincoli progettuali e tecnologici particolarmente rigidi, dettati dalle esigenze e dagli standard aziendali. Questo può limitare la flessibilità del team e aumentare la complessità delle attività di sviluppo e validazione</li><li>• Alta complessità riguardante le tecnologie da utilizzare con AWS, OCR (Optical Character Recognition), e integrazione dell'AI.</li></ul>

## 6 C6 - Second Brain

### 6.1 Breve descrizione

### 6.2 Requisiti funzionali e non funzionali

### 6.3 Tecnologie proposte

### 6.4 Chiarimenti e colloqui con l'azienda

### 6.5 Interesse del team

### 6.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• L'integrazione degli LLM in un'applicazione di produttività personale (note-taking) è un campo molto interessante. Lavorare su questo progetto fornisce un'esperienza diretta con una tecnologia all'avanguardia.</li><li>• Ampio Spazio per la Progettazione e la Ricerca: Il capitolato non specifica esattamente come implementare le funzionalità, ma si concentra sul cosa. Questo lascia al team un'enorme libertà nelle scelte architetturali, nella progettazione dell'interfaccia utente (UI/UX) e, soprattutto, nell'ingegnerizzazione dei prompt per l'LLM, che è il cuore della "ricerca" del progetto.</li><li>• Zucchetti si offre esplicitamente di supportare il team nelle parti più complesse (es. configurazione API LLM) e metterà a disposizione gli LLM stessi. Questo riduce un potenziale grande ostacolo.</li></ul>	<ul style="list-style-type: none"><li>• Complessità e Ampiezza del Dominio: Integrare un LLM in un'applicazione è un compito non banale. La gestione degli stati, degli errori delle API, dei costi (se si usano modelli cloud) e della latenza può essere complessa e richiedere molto tempo per la sola fase di prototipazione.</li><li>• Dipendenza da Tecnologie Esterne (LLM): L'intera applicazione dipende dalla stabilità, disponibilità e costi del servizio LLM scelto. Un cambiamento nelle API o problemi di disponibilità del servizio potrebbero bloccare lo sviluppo o il funzionamento dell'app.</li><li>• Le tecnologie non risultano chiaramente definite e i requisiti di progetto sono troppo ampi. Inoltre, l'assenza di specifiche vincolanti e di requisiti opzionali chiaramente delineati comporta un'eccessiva libertà decisionale, che può generare incertezza nella pianificazione e aumentare il rischio di deviazioni dagli obiettivi progettuali.</li><li>• L'azienda propone lo sviluppo di un'applicazione web basata su HTML per la parte front-end. Tuttavia, il team di progetto ritiene che una soluzione fondata esclusivamente su HTML non sia tecnicamente adeguata, soprattutto considerando la disponibilità di framework moderni e più performanti, quali Angular o React, che garantirebbero maggiore scalabilità, manutenibilità e qualità dell'esperienza utente.</li></ul>

## 7 C7 - Sistema di acquisizione dati da sensori

### 7.1 Breve descrizione

### 7.2 Requisiti funzionali e non funzionali

### 7.3 Tecnologie proposte

### 7.4 Chiarimenti e colloqui con l'azienda

### 7.5 Interesse del team

### 7.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• Il progetto affronta una sfida attuale e molto concreta nel campo IoT: acquisire, gestire e rendere fruibili dati da sensori in modo sicuro, scalabile e multi-tenant. Questo fornisce un'esperienza su problematiche industriali reali, non solo accademiche</li><li>• supporto??</li><li>• Forte opportunità di apprendimento: il progetto introduce tecnologie e pratiche non ancora esplorate dal team, accelerando la crescita tecnica. Nello specifico, il sistema di acquisizione dati da sensori si basa su un'architettura a microservizi e introduce tecnologie come Node.js e Nest.js con TypeScript (backend) , Go (componenti ad alte prestazioni) , Angular (frontend SPA) , MongoDB (database NoSQL) , PostgreSQL (database SQL) , Redis (caching) , Google Cloud Platform (infrastruttura cloud) , Kubernetes (orchestrazione container) , NATS o Kafka (comunicazione asincrona) , e Prometheus con Grafana (monitoring)</li></ul>	<ul style="list-style-type: none"><li>• Progettare e implementare un'architettura distribuita, sicura e multi-tenant è una sfida tecnica notevole. La gestione della concorrenza, della consistenza dei dati, della comunicazione asincrona e della tolleranza ai guasti richiede competenze solide e un'attenta progettazione.</li><li>• I requisiti sono molti e articolati (dalle API on-demand e streaming, alla UI, al monitoring, alla sicurezza). C'è il concreto rischio di dover sacrificare la profondità di alcune funzionalità per coprirne la quantità entro i tempi del progetto.</li><li>• Testare un'architettura distribuita è intrinsecamente difficile. Configurare ambienti di test end-to-end, simulare guasti e testare la scalabilità richiederà sforzi notevoli e una buona pianificazione.</li><li>• Il gateway e i dispositivi devono essere simulati. I dati grezzi che vengono generati dai sensori devono essere realistici. Inoltre deve essere posta particolare attenzione alla coerenza (p.es i dati delle temperature devono essere non eccessive e adatte al contesto). Eventualmente i dati devono essere normalizzati. Rispetto al maggiore competitor (Vimar), M31 non prevede utilizzo di veri sensori per aiutare il team nello sviluppo.</li></ul>

## 8 C8 - Smart Order

### 8.1 Breve descrizione

*Analisi multimodale per la creazione automatica di ordini*, proposto da Ergon Informatica Srl, prevede la realizzazione di una piattaforma intelligente in grado di ricevere, interpretare e strutturare automaticamente ordini provenienti da input testuali, vocali e visivi, trasformando dati non strutturati in ordini cliente completi pronti per l'inserimento nei sistemi gestionali aziendali. L'obiettivo del progetto è quello di migliorare l'efficienza dei processi aziendali, riducendo al minimo l'intervento umano nelle fasi più ripetitive e soggette a errore.

## 8.2 Requisiti funzionali e non funzionali

## 8.3 Tecnologie proposte

## 8.4 Chiarimenti e colloqui con l'azienda

## 8.5 Interesse del team

Il gruppo ha valutato positivamente la modernità e la rilevanza del tema, che combina più ambiti dell'intelligenza artificiale offrendo l'opportunità di approfondire tecniche avanzate di Machine Learning e integrazione multimodale. L'architettura proposta è risultata flessibile e ben strutturata, permettendo l'uso di tecnologie moderne come LLM, OCR e sistemi speech-to-text, oltre a framework web come React o Blazor. È stato inoltre ritenuto molto positivo il supporto previsto da parte dell'azienda proponente e la possibilità di lavorare su un caso di studio reale fornito da Ergon.

## 8.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• L'integrazione di NLP, Computer Vision e Speech-to-Text in un'unica pipeline è un campo di ricerca e sviluppo estremamente attuale e complesso. Offre un'esperienza di apprendimento su tecnologie AI d'avanguardia.</li><li>• Viene fornito un ricco elenco di tecnologie suggerite per ogni componente (LLM, OCR, Speech-to-Text, UI), ma con la libertà di scegliere alternative. Questo permette al team di adattare lo stack tecnologico alle proprie competenze.</li></ul>	<ul style="list-style-type: none"><li>• Integrare e far collaborare modelli di AI diversi (LLM, Vision, Audio) in una pipeline coerente è una sfida tecnica notevolissima.</li><li>• Il team dovrebbe padroneggiare o imparare rapidamente una vasta gamma di tecnologie AI e di framework, ciascuna con le sue complessità.</li><li>• Il pre-processing di dati multimodali (pulizia del testo, elaborazione di immagini, trascrizione audio) è un compito laborioso e critico. La qualità dell'output finale dipende fortemente da questa fase.</li><li>• I modelli di AI, specialmente gli LLM, possono essere imprevedibili. Validare l'accuratezza e l'affidabilità del sistema in scenari reali, dove un errore può significare un ordine sbagliato, è una sfida significativa.</li></ul>

## 9 C9 - View4Life

### 9.1 Breve descrizione

Il capitolato proposto da Vimar S.p.A. ha come obiettivo la realizzazione di una piattaforma per la gestione intelligente di impianti domotici. all'interno di residenze protette per anziani, con l'intento di migliorare la sicurezza, il comfort e l'efficienza energetica degli ambienti. Il progetto prevede lo sviluppo di un'infrastruttura cloud e di un'applicazione web responsive dedicata al personale sanitario per il monitoraggio e il controllo dei dispositivi

### 9.2 Requisiti funzionali e non funzionali

### 9.3 Tecnologie proposte

### 9.4 Chiarimenti e colloqui con l'azienda

### 9.5 Interesse del team

Il gruppo ha valutato molto positivamente questo capitolato per la concretezza e l'impatto sociale: lavorare su un sistema capace di migliorare la qualità della vita e la sicurezza degli anziani rappresenta una forte motivazione, dando al progetto un valore tangibile e significativo. Sono stati inoltre apprezzati l'utilizzo di tecnologie moderne e versatili, come Docker, AWS e framework web come Angular o React, che offrono ampie opportunità di apprendimento e applicazione pratica. Un ulteriore punto di forza è rappresentato dal supporto fornito da Vimar, che garantisce incontri di avanzamento regolari e fornisce un kit hardware fisico per testare l'integrazione con dispositivi reali, riducendo il divario tra teoria e pratica e permettendo di affrontare concretamente le sfide dell'IoT.

La principale complessità del progetto risiede nella sua ampiezza e completezza, che richiede una pianificazione accurata e una buona capacità di prioritizzazione. Tuttavia, il gruppo ha interpretato questa caratteristica non come una difficoltà, ma come un'opportunità di crescita: gestire un progetto così articolato permetterà infatti di sviluppare competenze di organizzazione, coordinamento e gestione tecnica avanzata

### 9.6 Punti di forza e di debolezza

Punti di forza	Punti di debolezza
<ul style="list-style-type: none"><li>• Dominio Applicativo Concreto e Socialmente Utile: Il contesto delle "residenze protette" fornisce uno scopo nobile e tangibile. Sviluppare un sistema che può migliorare la sicurezza e il benessere degli anziani è un forte motivatore e rende il progetto molto più significativo rispetto a un dominio astratto.</li><li>• Tecnologie Moderne e Ricercate</li><li>• Supporto Aziendale Eccezionale e Materiale Fornito: Vimar fornisce un supporto strutturato con incontri bisettimanali/settimanali (SAL) e, aspetto cruciale, fornisce un kit hardware fisico per testare con dispositivi reali. Questo riduce enormemente il gap tra teoria e pratica e permette di affrontare problematiche reali di integrazione IoT.</li></ul>	<ul style="list-style-type: none"><li>• Elevata Complessità e Ampio Scope: La completezza del progetto è anche la sua principale sfida. Il team deve essere bravo a gestire la complessità e a prioritizzare le funzionalità obbligatorie, per non perdersi negli optional. Una buona pianificazione iniziale è fondamentale.</li><li>• Punto di debolezza 2</li><li>• Punto di debolezza 3</li></ul>