



UNIVERSITÀ DEGLI STUDI DI PADOVA  
CORSO DI LAUREA IN INFORMATICA (L-31)

Corso di Ingegneria del Software  
Anno Accademico 2025/2026

## Valutazione Capitolati

Redattori: Mihaela Mariana Romascu; Giovanni Ponso; Davide Biasuzzi  
Verificato da: Michele Ogniben

**Gruppo: NightPRO**

[swe.nightpro@gmail.com](mailto:swe.nightpro@gmail.com)

Data: 28 ottobre 2025

Versione: 0.4

**Tabella delle Versioni**

Versione	Data	Autore/i	Descrizione delle Modifiche
0.1	28/10/2025	Mihaela Mariana Romanescu; Giovanni Ponso; Davide Biasuzzi	Creazione bozza iniziale e struttura del documento.
0.2	28/10/2025	Davide Biasuzzi	Aggiunta Analisi per ogni capitolato, completata la conclusione
0.3	29/10/2025	Davide Biasuzzi	Corretto refuso sulla data
0.4	30/10/2025	Giovanni Ponso	Aggiunta contenuti e revisione finale

## Indice

<b>Tabella delle Versioni</b>	<b>2</b>
<b>1 Informazioni Generali</b>	<b>4</b>
1.1 Componenti del Gruppo . . . . .	4
<b>2 C1. Automated EN18031 Compliance Verification</b>	<b>5</b>
<b>3 C2. Code Guardian</b>	<b>6</b>
<b>4 C3. DIPReader</b>	<b>7</b>
<b>5 C4. L'app che Protegge e Trasforma</b>	<b>8</b>
<b>6 C5. NEXUM</b>	<b>9</b>
<b>7 C6. Second Brain</b>	<b>10</b>
<b>8 C7. Sistema di acquisizione dati da sensori</b>	<b>11</b>
<b>9 C8. Smart Order</b>	<b>12</b>
<b>10 C9. View4Life</b>	<b>13</b>
<b>11 Conclusione e Scelta Finale del Gruppo NightPRO</b>	<b>14</b>

## 1 Informazioni Generali

### 1.1 Componenti del Gruppo

Cognome	Nome	Matricola
Biasuzzi	Davide	2111000
Bilato	Leonardo	2071084
Zanella	Francesco	2116442
Romascu	Mihaela-Mariana	2079726
Ogniben	Michele	2042325
Perozzo	Samuele	2110989
Ponso	Giovanni	2000558

Tabella 1: Componenti del gruppo NightPRO.

## 2 C1. Automated EN18031 Compliance Verification

### Valutazione del capitolo

Il progetto proposto da Bluewind prevede lo sviluppo di un'applicazione per assistere la verifica di conformità alla norma EN18031 (Direttiva RED), digitalizzando i *decision tree* della normativa e guidando l'utente nel processo di valutazione. È previsto un caso d'uso concreto (macchina del caffè connessa), utile per testare sicurezza e autenticazione su dispositivi IoT.

Il capitolo fornisce materiale reale, indica Python come tecnologia preferenziale e garantisce supporto diretto da parte dell'azienda.

#### Pro

- Tema reale legato alla sicurezza IoT e alla conformità normativa.
- Processo chiaro e strutturato grazie ai decision tree.
- Caso studio concreto fornito dall'azienda.
- Supporto tecnico costante e buona libertà tecnologica.

#### Contro

- Curva iniziale di apprendimento sulla normativa EN18031.
- Modellazione dei decision tree complessa e centrale per il progetto.
- Alto focus su struttura dati e parsing, meno su UI/UX.

#### Analisi

La difficoltà principale riguarda la rappresentazione dei *decision tree* e la costruzione di una struttura dati flessibile per interpretarli. Conviene partire dalla definizione del modello e dal parser, rimandando la GUI a una fase successiva. Un progetto interessante e ben guidato, ma con un carico iniziale legato alla comprensione della normativa.

### 3 C2. Code Guardian

#### Valutazione del capitolo

Il progetto prevede la realizzazione di una piattaforma in grado di analizzare repository GitHub e fornire valutazioni sulla qualità del codice, sicurezza e manutenzione. L'idea è supportare sviluppatori e team DevOps tramite agenti specializzati che producono report automatici e suggerimenti di miglioramento. L'architettura è basata su un orchestratore che coordina più agenti, con una dashboard web per visualizzare i risultati.

La proposta è interessante e attuale, soprattutto per chi vuole approfondire strumenti e metodologie legate alla qualità del software e alla sicurezza. Il capitolo richiede l'uso di tecnologie moderne (Node.js, Python, React) e introduce anche aspetti di CI/CD e analisi di sicurezza secondo standard come OWASP. Inoltre, è previsto supporto costante da parte dell'azienda.

#### Pro

- Tema moderno e rilevante per l'ambito DevOps e software engineering.
- Architettura modulare basata su agenti, che permette un lavoro strutturato e incrementale.
- Utilizzo di tecnologie attuali.
- Buone opportunità di apprendimento su sicurezza, testing e analisi del codice.
- Supporto e mentoring tecnico da parte dell'azienda.

#### Contro

- Progetto complesso, soprattutto nella gestione degli agenti e della loro comunicazione.
- Ampiezza dell'ambito, rischio di voler coprire troppo in poco tempo.
- Necessità di definire metriche chiare per valutare codice e sicurezza.
- Richiede familiarità con più tecnologie e concetti (backend, frontend, DevOps, OWASP).

#### Analisi

La principale difficoltà è l'ampiezza del progetto e la complessità dell'architettura multi-agente. Per ridurre il rischio, avrebbe senso iniziare con un MVP limitato, implementando l'orchestratore e un singolo agente (ad esempio per l'analisi dei test o della sicurezza) e poi espandere in modo graduale. Nel complesso si tratta di un capitolo formativo, ma impegnativo dal punto di vista tecnico e organizzativo.

## 4 C3. DIPReader

### Valutazione del capitolo

Il progetto punta a realizzare un software multipiattaforma per la consultazione e ricerca di archivi digitali provenienti da sistemi di conservazione documentale.

L'uso di SQLite e FAISS rende il sistema leggero ma moderno, con possibilità di integrare ricerca semantica. Il capitolo fornisce un MVP ben definito e richiede che l'app funzioni senza installazioni esterne, elementi che guidano in modo chiaro sia l'architettura che le priorità iniziali.

#### Pro

- Tema concreto e utile per aziende e PA.
- MVP chiaro e realistico.
- Tecnologie leggere e diffuse.
- Funzionamento offline e multipiattaforma.
- Supporto aziendale solido e materiale reale.

#### Contro

- Necessità di ottimizzare prestazioni su grandi volumi.
- Ricerca semantica complessa da implementare.
- Comprensione minima del contesto normativo richiesta.
- UI/UX da progettare con cura per evitare eccessiva tecnicità.

#### Analisi

Il punto critico è la Data Pipeline di Ingestion. Passare dal pacchetto .zip a un database SQLite interrogabile in modo efficiente. Il focus progettuale deve essere sul parsing degli XML e sulla normalizzazione dei dati ancora prima di iniziare a disegnare l'UI.

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

### Valutazione del capitolo

Il progetto prevede lo sviluppo di un'app mobile dedicata al supporto e alla prevenzione di situazioni di violenza di genere. L'obiettivo è offrire uno strumento sicuro e discreto che permetta alle persone a rischio di chiedere aiuto in modo rapido e non tracciabile, raccogliere prove digitali criptate e contribuire a percorsi di supporto.

La soluzione integra funzionalità di riconoscimento del linguaggio tramite AI, alert silenziosi, geolocalizzazione sicura e gestione di un diario cifrato.

#### Pro

- Forte valore sociale e impatto reale.
- Tecnologie moderne (AI, AWS, Flutter).
- Supporto tecnico e formativo continuo.
- Focus su sicurezza, privacy e inclusività.

#### Contro

- Alta complessità tecnica e gestionale.
- Rischio di dispersione se non si limita l'MVP.
- Implementazione delicata della privacy.
- Richiede sensibilità etica oltre che tecnica.

#### Analisi

Il progetto richiede una sensibilità particolare sul tema sociale trattato, inoltre è ad altissimo rischio per quanto riguarda la responsabilità sui dati. Richiede un approccio complesso "Security by Design" fin dall'inizio. La sfida tecnica principale è la gestione delle chiavi di crittografia per il "diario criptato".

## 6 C5. NEXUM

### Valutazione del capitolo

Il progetto NEXUM riguarda lo sviluppo di funzionalità avanzate per una piattaforma HR già esistente, con l'obiettivo di digitalizzare e semplificare attività come gestione documentale, distribuzione dei cedolini e comunicazioni aziendali. Il capitolo prevede l'integrazione di moduli basati su AI generativa e OCR per automatizzare parte del lavoro amministrativo, mantenendo comunque un controllo umano sulle correzioni.

Il contesto d'uso è concreto e legato a processi reali di aziende e studi di consulenza del lavoro, con la possibilità di lavorare su casi pratici e flussi che andranno effettivamente in produzione.

#### Pro

- Progetto concreto e destinato al mercato reale.
- Innovazione con AI generativa e OCR.
- Stack moderno e formazione Agile.
- Supporto costante e ambiente professionale.

#### Contro

- Alta complessità tecnica e integrazione AI.
- Requisiti di precisione elevati.
- Dipendenza da servizi esterni (OCR, LLM).

#### Analisi

Il successo non dipende dalla precisione del 100% dell'AI, ma dall'efficienza dell'interfaccia di correzione: serve rendere rapida la validazione dei documenti tramite flusso "human-in-the-loop". La sfida principale è progettare un'interfaccia che minimizzi i tempi di revisione. Inoltre data la grande quantità di funzionalità previste è necessario definire un MVP limitato e concentrarsi su un solo processo (es. upload → OCR → validazione).

## 7 C6. Second Brain

### Valutazione del capitolo

Il progetto Second Brain prevede lo sviluppo di un editor Markdown capace di integrare funzionalità LLM per supportare la scrittura, la rielaborazione dei contenuti e attività di brainstorming. L'obiettivo è sperimentare come l'AI possa potenziare la produttività personale e creativa, permettendo all'utente di migliorare testi, generarne di nuovi o analizzarli da diverse prospettive.

#### Pro

- Tema innovativo e attuale.
- Progetto scalabile e sperimentale.
- Requisiti chiari e raggiungibili.
- Supporto tecnico disponibile.

#### Contro

- Limitato impatto industriale immediato.
- Prompt Engineering complesso.
- Possibili problemi di privacy con API esterne.

#### Analisi

La sfida non è l'architettura (un editor Markdown che chiama API), ma il Prompt Engineering. Il successo dipende dalla qualità dei prompt per i "sei capelli". E un progetto affascinante ma con meno complessità architetturale.

## 8 C7. Sistema di acquisizione dati da sensori

### Valutazione del capitolo

Il progetto propone la realizzazione della componente cloud di un sistema IoT per raccogliere e gestire dati provenienti da sensori BLE tramite gateway simulati. L'obiettivo è costruire un'infrastruttura scalabile e sicura che gestisca autenticazione dei nodi, segregazione multi-tenant e flussi di dati in tempo reale, fornendo anche API e una UI base per la consultazione.

L'attenzione è rivolta alla progettazione di un backend distribuito e orientato a microservizi, con focus su sistemi di messaggistica asincrona (message queues) e gestione dei flussi di eventi..

#### Pro

- Architettura cloud-IoT ben strutturata, simile a quella usata in contesti reali.
- Focus su sicurezza e scalabilità.
- Tecnologie moderne e richieste dal mercato.
- Ottima esperienza formativa.

#### Contro

- Elevata complessità tecnica e di setup.
- Limitata componente frontend.
- Rischio di tempi lunghi di sviluppo.

#### Analisi

Questo capitolo è una pura sfida di Backend Asincrono e Multi-Tenant. Richiede la simulazione di gateway multipli che inviano dati in parallelo. Il cuore del progetto è l'architettura a code e la corretta segnalazione dei dati per tenant, molto prima di pensare alla dashboard.

## 9 C8. Smart Order

### Valutazione del capitolo

Il progetto Smart Order prevede la realizzazione di una piattaforma in grado di ricevere ordini provenienti da canali diversi (testo, audio e immagini) e convertirli automaticamente in ordini strutturati pronti per l'integrazione con sistemi ERP aziendali. L'obiettivo è automatizzare la fase di interpretazione e normalizzazione degli ordini cliente, riducendo errori e tempi di inserimento manuale.

Il sistema si basa su una pipeline completa che combina NLP, riconoscimento vocale e OCR, integrati tramite modelli AI multimodali e validazione da parte dell'operatore. Il progetto bilancia sperimentazione su tecnologie avanzate e applicazione a un caso industriale reale, con la possibilità di lavorare su dati forniti dall'azienda.

#### Pro

- Innovativo e concreto.
- Architettura modulare e scalabile.
- Possibilità di scegliere componenti e modelli AI tra open-source e servizi cloud
- Supporto aziendale solido.

#### Contro

- Rischio di dispersione data la natura multimodale
- La gestione delle ambiguità negli ordini e dei casi non standard richiede logiche solide di triage e supervisione.
- Miglioramento automatico del modello basato sui feedback: concetto potente ma non banale da implementare in modo affidabile.
- Necessita coordinamento tra pipeline AI, backend e interfaccia per la validazione umana.

#### Analisi

Il progetto richiede di costruire un flusso completo: acquisizione degli ordini (in varie forme), estrazione delle informazioni, interpretazione tramite modelli AI e validazione da parte dell'operatore prima dell'invio al sistema gestionale. La complessità è quindi distribuita su più livelli, ma la struttura modulare consente di suddividere il lavoro tra i membri del team e procedere per fasi.

Per gestire il rischio di dispersione può essere utile concentrarsi inizialmente su una sola modalità (ad esempio testo), estendendo poi ad audio e immagini solo dopo aver consolidato il flusso principale.

Riteniamo ideale la natura modulare del progetto per un team numeroso come il nostro.

## 10 C9. View4Life

### Valutazione del capitolo

Il progetto proposto da Vimar, prevede lo sviluppo di una piattaforma cloud e web per la gestione di impianti domotici installati in residenze protette per anziani. L'obiettivo è fornire uno strumento unico per il personale sanitario, in grado di monitorare ambienti, sensori e dispositivi, gestire allarmi e visualizzare dati dell'impianto in tempo reale tramite l'API KNX IoT.

L'azienda fornisce un kit hardware reale per la sperimentazione e l'integrazione, rendendo il progetto concreto e collegato a scenari domotici reali.

#### Pro

- Progetto reale con finalità sociali.
- Tecnologie moderne (IoT, Cloud, API).
- Esperienza con dispositivi fisici.
- Supporto aziendale concreto.

#### Contro

- Gestione di logiche sensibili: distinguere eventi critici (es. cadute) da comportamenti normali richiede attenzione.
- Integrazione IoT non banale, con comunicazione real-time tramite KNX IoT API.
- Necessità di evitare falsi allarmi o mancate segnalazioni, dato il contesto delicato.
- Richiede coordinamento costante tra backend, API e interfaccia di controllo.

#### Analisi

Il progetto combina domotica reale, cloud e integrazione con dispositivi KNX. La presenza di hardware fisico è un valore aggiunto, ma implica una gestione attenta dell'integrazione: per accelerare lo sviluppo può essere utile predisporre un piccolo mock dell'API, così da lavorare anche senza il kit sempre disponibile.

La parte più delicata riguarda la gestione delle logiche di allarme in un contesto sensibile come quello delle residenze per anziani. È necessario definire in modo accurato quando notificare eventi critici e ridurre al minimo falsi positivi e falsi negativi. Questo rende il progetto stimolante, ma anche impegnativo in termini di responsabilità e validazione.

## 11 Conclusione e Scelta Finale del Gruppo NightPRO

L'analisi dei nove capitolati ha evidenziato un livello qualitativo complessivamente alto, con proposte che spaziano tra Intelligenza Artificiale, architetture cloud e sicurezza informatica.

La valutazione interna del gruppo **NightPRO**, basata su criteri di innovazione, fattibilità, chiarezza dei requisiti e valore formativo, ha portato a individuare due progetti finalisti: **C6 – Second Brain** e **C8 – SmartOrder**. Entrambi applicano modelli di AI a contesti reali, ma con livelli di complessità architetturale differenti.

### Analisi del "Runner-up": C6 – Second Brain

Il progetto **C6** di Zucchetti è risultato interessante per la chiarezza dell'MVP e per il focus sul **Prompt Engineering**. Tuttavia, la componente software di base è relativamente semplice: un editor Markdown che interagisce con modelli linguistici tramite API. Il progetto ha quindi un carattere più esplorativo che ingegneristico, orientato alla sperimentazione sull'uso dei LLM.

### Decisione finale: C8 – SmartOrder

Il gruppo **NightPRO** ha raggiunto un accordo comune sulla scelta del capitolato **C8 – SmartOrder**, proposto da **Ergon Informatica S.r.l.**

SmartOrder rappresenta una sfida più completa dal punto di vista dell'ingegneria del software. Oltre all'impiego di modelli AI, richiede la progettazione di una pipeline end-to-end composta da:

- **Ingestion:** ricezione di input non strutturati (testo inizialmente, in futuro audio e immagini)
- **Pre-processing:** normalizzazione dei dati tramite tecniche NLP e OCR
- **AI Core:** estrazione di entità (prodotti, quantità, riferimenti) e classificazione degli ordini
- **Validation (Human-in-the-Loop):** gestione delle ambiguità e conferma manuale degli ordini
- **Output Generation:** produzione di un ordine strutturato (JSON) per l'integrazione ERP

Questa architettura modulare favorisce una chiara suddivisione delle responsabilità tra i membri del gruppo e supporta un approccio incrementale allo sviluppo.

L'incontro con il Dott. Carlesso di Ergon ha inoltre ridotto il principale rischio tecnico, confermando la possibilità di limitare l'MVP iniziale alla sola modalità testuale. Questo permette di gestire la complessità multimodale in modo progressivo.

Per questi motivi, **SmartOrder** è stato scelto come progetto finale del gruppo: un capitolato innovativo, tecnicamente stimolante e con un percorso di crescita concreto e bilanciato per tutti i membri.