



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

Corso di Ingegneria del Software  
Anno Accademico 2025/2026

# Valutazione Capitolati

Redattori: Romascu Mihaela Mariana; Ponso Giovanni; Biasuzzi Davide  
Verificato da: Tutto il gruppo

**Gruppo: NightPRO**

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

Data: 28 ottobre 2025

Versione: 0.3

## Indice

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

Il progetto di **Bluewind** mira a realizzare un'applicazione software (desktop o web) per automatizzare la verifica di conformità alla norma EN18031, relativa alla sicurezza informatica dei dispositivi radio secondo la Direttiva RED.

L'idea risponde a un'esigenza reale: ridurre la complessità e gli errori del processo manuale di verifica normativa.

La soluzione digitalizza i *Decision Tree* della norma, consentendo all'utente di navigarli, ottenere risultati automatici ("Pass", "Fail", "Not applicable") e visualizzarli in una dashboard interattiva.

Il caso di studio scelto — una macchina del caffè connessa via Wi-Fi — è realistico e utile per testare autenticazione, scambio sicuro di dati e logiche di verifica.

Il capitolato offre flessibilità tecnologica (Python consigliato, ma non vincolante) e un approccio Agile, favorendo autonomia e adattabilità. Tuttavia, la complessità normativa e la scarsa specificità tecnica del documento possono rendere impegnativa la realizzazione per studenti senza conoscenze in ambito cybersecurity o standard europei.

### Pro

- Tema reale e utile nel contesto della sicurezza IoT.
- Automazione di un processo complesso e ripetitivo.
- Libertà di scelta tecnologica.
- Supporto aziendale costante.
- Caso di test concreto e chiaro.
- Approccio Agile e flessibile.

### Contro

- Comprensione complessa della norma EN18031.
- Mancanza di dettagli su architettura e parsing dei Decision Tree.
- Carico cognitivo elevato nella modellazione dei flussi decisionali.
- Scarsa chiarezza sui criteri di valutazione automatica.
- UI/UX poco approfondita nel documento.

### Analisi

Il Rischio non è l'UI, ma la modellazione dei dati. Il successo dipende dalla creazione di uno schema (e.g.: JSON) che possa rappresentare qualsiasi albero decisionale della norma EN18031. La prima milestone dovrebbe essere un parser di questo schema, non la GUI.

## 3 C2. Code Guardian

### Valutazione del Capitolato

Il progetto propone una piattaforma intelligente per analizzare repository GitHub, valutando qualità del codice, sicurezza e manutenzione, e fornendo report e suggerimenti di miglioramento. L'architettura prevede un orchestratore centrale che coordina agenti specializzati (per test, sicurezza, documentazione, report), con una dashboard web in React.

#### Pro

- Tema innovativo e rilevante per il mondo DevOps.
- Architettura moderna e modulare con AI e multi-agenti.
- Stack tecnologico attuale (Node.js, Python, React, AWS).
- Supporto aziendale professionale e continuo.
- Forte componente formativa in sicurezza e qualità del software.

#### Contro

- Elevata complessità tecnica e di integrazione.
- Ambito vasto, rischio di dispersione.
- Definizione non chiara delle metriche di valutazione.
- Tempi lunghi per ottenere una demo completa.

#### Analisi

Riteniamo sia presente un alto rischio di dispersione. L'architettura multi-agente è complessa. Una mitigazione efficace sarebbe un MVP iper-limitato: costruire l'orchestratore e un solo agente (e.g.: "Test Analysis") ignorando tutti gli altri per la prima consegna.

## 4 C3. DIPReader

### Valutazione del Capitolato

Il progetto punta a realizzare un software multiplatforma 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.

#### Pro

- Tema concreto e utile per aziende e PA.
- MVP chiaro e realistico.
- Tecnologie leggere e diffuse.
- Funzionamento offline e multiplatforma.
- 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 file .zip a un db SQLite interrogabile in modo performante. 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 Capitolato**

Il progetto propone un'app mobile per prevenire e gestire situazioni di violenza di genere, con funzionalità di analisi AI, alert silenziosi e geolocalizzazione sicura.

**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**

Progetto 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": la chiave deve essere gestita solo sul dispositivo dell'utente, altrimenti la privacy è compromessa.

## 6 C5. NEXUM

**Valutazione del Capitolato**

Il progetto NEXUM punta a digitalizzare la gestione HR nelle PMI, migliorando la comunicazione tra aziende, consulenti e dipendenti, con moduli basati su AI generativa e OCR.

**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'UI di correzione, Il progetto si vince progettando un interfaccia che permetta all'operatore di validare i documenti in poco tempo.

## 7 C6. Second Brain

**Valutazione del Capitolato**

Il progetto Zucchetti propone un editor di testo intelligente che utilizza LLM per assistere nella scrittura, revisione e generazione di testi, basato su Markdown.

**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 architettureale.

## 8 C7. Sistema di acquisizione dati da sensori

### Valutazione del Capitolato

Il progetto prevede la creazione di un'infrastruttura cloud per la gestione di dati da sensori BLE, simulando un ecosistema IoT.

#### Pro

- Architettura coerente e professionale.
- 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 capitolato 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 Capitolato

Il progetto propone una piattaforma multimodale intelligente capace di interpretare ordini d'acquisto provenienti da testo, audio e immagini, trasformandoli in ordini strutturati per ERP.

#### Pro

- Innovativo e concreto.
- Architettura modulare e scalabile.
- Libertà tecnologica.
- Supporto aziendale solido.

#### Contro

- Alta complessità tecnica.
- Necessità di dati di qualità per training.
- Integrazione ERP complessa.



## Analisi

Smart Order è una Sfida di Pipeline Architettuale Completa. A differenza di altri richiede di progettare un flusso end-to-end:

1. Ingestion (API/Web)
2. Pre-processing (NLP/OCR)
3. AI Core (Matching)
4. Validazione (UI per l'operatore)
5. Output (JSON/DB)

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

## 10 C9. View4Life

### Valutazione del Capitolato

Il progetto Vimar prevede una piattaforma cloud e web per la gestione di impianti domotici nelle residenze per anziani, basata su KNX IoT API.

#### Pro

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

#### Contro

- Alta complessità tecnica e di integrazione.
- Rischio logistico legato ai dispositivi fisici.
- Necessario coordinamento costante del team.

## Analisi

Questo progetto offre un vantaggio unico: validazione su un kit HW reale fornito dall'azienda. Tuttavia lo sviluppo dell'applicativo WEB dipende da un'API esterna (KNX IoT) e dall'accesso a questo HW, che potrebbe diventare un bottleneck per un team numeroso. Sarebbe necessario sviluppare un "Mock Server" che faccia da simulatore replicando le risposte attese dall'API KNX IoT, in modo da poter testare la piattaforma senza dipendere dall'HW fisico, che verrà poi utilizzato nella fase finale di integrazione e validazione.

## 11 Conclusione e Scelta Finale del Gruppo NightPRO

L'analisi complessiva dei nove capitolati ha rivelato un livello qualitativo elevato, con numerose opportunità di approfondimento in ambiti di frontiera come l'Intelligenza Artificiale, le architetture cloud e la sicurezza.

La valutazione interna del gruppo **NightPRO**, basata su criteri di innovazione tecnologica, fattibilità, chiarezza dei requisiti e valore formativo, ha portato all'identificazione di due finalisti: **C6 – Second Brain** e **C8 – SmartOrder**.

Entrambi i progetti sono di grande interesse, focalizzati sull'applicazione di modelli AI (LLM e NLP) a problemi concreti. Tuttavia, un'analisi ingegneristica più approfondita ha evidenziato differenze sostanziali che hanno guidato la nostra decisione finale.

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

Il progetto **C6** di Zucchetti è affascinante e presenta requisiti d'MVP (Minimum Viable Product) estremamente chiari. La sfida principale risiede quasi esclusivamente nel **Prompt Engineering**, ovvero nello sviluppare la logica per far sì che un LLM generi critiche efficaci secondo il modello dei "sei cappelli per pensare".

Tuttavia, l'architettura software di base (un editor Markdown locale che effettua chiamate API) è relativamente semplice. Il progetto ha uno spiccato "carattere esplorativo", più simile a un prototipo di Ricerca e Sviluppo che a un sistema software ingegneristicamente complesso.

### Motivazione della Scelta: C8 – SmartOrder

Il gruppo **NightPRO**, ha scelto all'unanimità il capitolato **C8 – SmartOrder** di Ergon Informatica Srl.

Riteniamo che questo progetto rappresenti una **sfida di Ingegneria del Software più completa e robusta**. A differenza di C6, il successo di SmartOrder non dipende solo dalla qualità del modello AI, ma dalla progettazione di un'intera **pipeline architetturale modulare**, che include:

- **Ingestion:** Un'interfaccia per ricevere input non strutturati (testo, audio, immagini).
- **Processing:** Moduli di pre-elaborazione e AI per estrarre entità (articoli, quantità).
- **Validazione:** Una logica di "parcheggio" per gli ordini ambigui, che richiede un'interfaccia *Human-in-the-Loop* (HITL) per la validazione manuale.
- **Output:** La generazione di un ordine strutturato (es. JSON) per i sistemi ERP.

Questa architettura *end-to-end* si presta magnificamente alla parallelizzazione del lavoro per un team numeroso come il nostro, permettendo la creazione di un sistema complesso e integrato.

Inoltre, l'incontro chiarificatore con il Dott. Carlesso di Ergon (come riportato nella nostra Lettera di Presentazione) ha efficacemente **mitigato il rischio tecnico principale** (la complessità multimodale), confermando che l'MVP può focalizzarsi sulla sola modalità testuale.

Per questi motivi, **SmartOrder** offre il miglior equilibrio tra innovazione (AI, NLP), complessità architetturale e gestione realistica del progetto.