



UNIVERSITÀ DEGLI STUDI DI PADOVA
CORSO DI LAUREA IN INFORMATICA (L-31)
Corso di Ingegneria del Software
Anno Accademico 2025/2026

Piano di Qualifica

Redattori: Mihaela-Mariana Romascu, Davide Biasuzzi

Gruppo: NightPRO
swe.nightpro@gmail.com

Data: 2025-12-18

Versione: 0.2

Tabella delle Versioni

Versione	Data	Autore/i	Descrizione	Verificatore
0.1	2025-12-05	Mihaela-Mariana Roma-scu	Stesura iniziale del documento, definizione metriche di processo e prodotto.	Giovanni Ponso
0.2	2025-12-18	Davide Biasuzzi	Aggiunto paragrafo relativo alla qualità della documentazione	Samuele Perozzo

Indice

Tabella delle Versioni	2
Informazioni Generali	4
1 Introduzione	5
1.1 Scopo del documento	5
1.2 Glossario	5
2 Riferimenti	5
2.1 Riferimenti normativi	5
2.2 Riferimenti informativi	5
3 Qualità di Processo	6
3.1 Scopo e Obiettivi	6
3.2 Parametri del Progetto	6
3.3 Metriche di Gestione del Progetto	6
3.3.1 Definizione Metriche	6
3.3.2 Valori di Accettazione	7
4 Qualità di Prodotto	8
4.1 Metriche di Qualità del Software (Statica)	8
4.2 Metriche di Qualità del Software (Dinamica)	8
5 Qualità della Documentazione	9
5.1 Metriche di Leggibilità e Correttezza Linguistica	9
5.1.1 Indice Gulpease	9
5.1.2 Controllo Grammaticale e Linguistico (LanguageTool)	9
5.1.3 Controllo Sintattico L ^A T _E X (ChkTeX)	9
5.2 Soglie di Accettazione e Target	10
5.3 Grafici	10
5.3.1 Indice Gulpease	10
5.3.2 LanguageTool	11
5.3.3 ChkTeX	11
5.4 Automazione e Verifica Continua	11

Informazioni Generali

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.

1 Introduzione

Il *Piano di Qualifica* rappresenta il documento strategico per la gestione della qualità all'interno del progetto **SmartOrder**. Esso stabilisce gli standard, le metodologie di verifica e le metriche quantitative necessarie per valutare sia la qualità del processo di sviluppo che la qualità del prodotto software finale.

1.1 Scopo del documento

Questo documento ha lo scopo di definire:

- Gli **obiettivi di qualità** che il gruppo NightPRO si impegna a raggiungere.
- Le **metriche** utilizzate per monitorare l'andamento del progetto (costi, tempi) e la qualità del codice.
- Le **procedure di verifica e validazione** da applicare ai documenti e al software.
- I criteri di accettazione per le fasi di avanzamento.

L'adozione di questo piano garantisce che il prodotto finale sia conforme ai requisiti espressi nel Capitolato C8 e rispetti gli standard accademici richiesti.

1.2 Glossario

Al fine di evitare ambiguità, i termini tecnici o specifici del dominio sono raccolti nel documento *Glossario*. Nel presente testo, tali termini sono contrassegnati con una "G" in pedice alla loro prima occorrenza (es. Agile_G).

2 Riferimenti

2.1 Riferimenti normativi

- [Norme di Progetto V1.0](#)
- [Capitolato d'appalto C8 - Smart Order](#)
- [Capitolato d'appalto C8 - Smart Order](#)

2.2 Riferimenti informativi

- [Glossario](#)
- [Analisi dei requisiti \[da inserire link\]](#)
- [Verbali esterni](#)
- [Verbali interni](#)
- [Piano di progetto](#)

3 Qualità di Processo

3.1 Scopo e Obiettivi

La qualità di processo riguarda la capacità del gruppo di gestire le attività pianificate rispettando vincoli di budget e tempistiche. Il gruppo NightPRO adotta il ciclo di Deming, noto come PDCA:

- **Plan (Pianificare):** stabilire gli obiettivi e i processi necessari per fornire risultati in accordo con i risultati attesi.
- **Do (Fare):** attuare il piano, eseguire i processi, realizzare il prodotto.
- **Check (Verificare):** monitorare e misurare i processi e il prodotto a fronte delle politiche, degli obiettivi e dei requisiti.
- **Act (Agire):** adottare azioni per migliorare continuamente le prestazioni dei processi.

3.2 Parametri del Progetto

I seguenti valori costituiscono la baseline per il calcolo delle metriche:

Parametro	Descrizione	Valore
BAC (Budget At Completion)	Costo totale preventivato del progetto (v0.4)	€ 12.850,00
Ore Totali	Monte ore complessivo disponibile	630 ore
Componenti	Numero di membri del gruppo	7
Ore medie per componente	Media ore di lavoro pro capite (630/7)	90 ore
Scadenza Progetto	Data prevista per la consegna finale (PB)	21/03/2026

Tabella 2: Parametri fondamentali del progetto

3.3 Metriche di Gestione del Progetto

Per il monitoraggio dell'avanzamento, il gruppo utilizza la metodologia Earned Value Management.

3.3.1 Definizione Metriche

AC (Actual Cost) : Costo realmente sostenuto fino alla data corrente.

EV (Earned Value) : $EV = BAC \cdot (\% \text{ completamento attività})$

PV (Planned Value) : $PV = BAC \cdot (\% \text{ lavoro pianificato alla data corrente})$

CV (Cost Variance) : $CV = EV - AC$ (Differenza di costo)

SV (Schedule Variance) : $SV = EV - PV$ (Differenza di tempi)

CPI (Cost Performance Index) : $CPI = \frac{EV}{AC}$ (Efficienza economica)

SPI (Schedule Performance Index) : $SPI = \frac{EV}{PV}$ (Efficienza temporale)

EAC (Estimate At Completion) : $EAC = AC + \frac{BAC - EV}{CPI}$ (Stima costo a finire)

3.3.2 Valori di Accettazione

Metrica	Descrizione	Range Ottimale	Accettabile
CV	Cost Variance	≥ 0	$\geq -5\%$ BAC
SV	Schedule Variance	≥ 0	$\geq -5\%$ PV
CPI	Cost Performance Index	≥ 1.00	$0.95 \leq CPI \leq 1.05$
SPI	Schedule Performance Index	≥ 1.00	$0.95 \leq SPI \leq 1.05$

Tabella 3: Soglie di accettazione per le metriche EVM

4 Qualità di Prodotto

Il gruppo NightPRO fa riferimento allo standard **ISO/IEC 25010** per definire le caratteristiche di qualità del prodotto software (es. Manutenibilità, Usabilità, Efficienza, Sicurezza).

4.1 Metriche di Qualità del Software (Statica)

Queste metriche mirano a garantire la manutenibilità e la qualità del codice sorgente prodotto.

CX (Cyclomatic Complexity) : $CX = E - N + 2P$ (Complessità del flusso di controllo)

DI (Depth of Inheritance) : Profondità massima dell'albero di ereditarietà

CD (Code Duplication) : Percentuale di codice duplicato sul totale

CCh (Code Churn) : Frequenza e volume delle modifiche al codice nel tempo

TD (Technical Debt) : Tempo stimato per il refactoring del codice problematico

Metrica	Descrizione	Ottimale	Accettabile
CX	Complessità Ciclomatica (per metodo)	≤ 10	≤ 15
DI	Profondità Ereditarietà	≤ 4	≤ 6
CD	Duplicazione Codice	0%	$\leq 5\%$
CCh	Code Churn (linee modificate/settimana)	Basso	Medio
TD	Debito Tecnico	≤ 2 gg	≤ 5 gg

Tabella 4: Metriche di Qualità del Codice

4.2 Metriche di Qualità del Software (Dinamica)

Per garantire la correttezza del software, vengono misurati i livelli di copertura dei test (*Code Coverage*).

CC (Code Coverage) : $\frac{\text{Linee di codice eseguite dai test}}{\text{Linee totali di codice}} \cdot 100\%$

BC (Branch Coverage) : $\frac{\text{Rami decisionali eseguiti}}{\text{Rami totali}} \cdot 100\%$

Indicatore	Descrizione	Target (PB)
CC	Copertura del codice (Linee)	$\geq 80\%$
BC	Copertura dei rami (Branch)	$\geq 70\%$

Tabella 5: Metriche di Verifica dinamica

5 Qualità della Documentazione

La documentazione accompagna il prodotto software e deve garantire accessibilità e comprensione a tutti gli stakeholder. La qualità della documentazione è verificata attraverso un insieme coordinato di strumenti automatizzati che controllano la leggibilità linguistica, la correttezza grammaticale e la sintassi \LaTeX .

5.1 Metriche di Leggibilità e Correttezza Linguistica

Per valutare la qualità dei documenti redatti in italiano, il gruppo utilizza un insieme di indici automatizzati, ognuno specifico per un aspetto della qualità documentale.

5.1.1 Indice Gulpase

L'indice Gulpase è calibrato specificamente per la lingua italiana e valuta la leggibilità di un testo basandosi sulla lunghezza delle parole e delle frasi. Un indice elevato indica un testo più facilmente comprensibile.

$$G = 89 + \frac{300 \cdot (\text{numero frasi}) - 10 \cdot (\text{numero lettere})}{\text{numero parole}}$$

L'indice viene calcolato automaticamente su tutti i file `.tex` presenti nel repository, escludendo da tale calcolo i comandi \LaTeX , le formule matematiche e i commenti. La verifica è eseguita mediante lo script `check_gulpase.py`, che genera un report CSV contenente il valore Gulpase per ogni documento.

5.1.2 Controllo Grammaticale e Linguistico (LanguageTool)

LanguageTool è uno strumento di verifica grammaticale che identifica errori ortografici, grammaticali e stilistici nei testi in italiano. La verifica automatica:

- Rileva errori di ortografia, concordanza verbale e nominale, punteggiatura;
- Segnala problemi di stile e ridondanze lessicali;
- Esclude dal controllo i nomi propri del gruppo, i termini tecnici presenti nel glossario e gli identificativi tecnici (branch, commit, tool name, ecc.).

Lo script `check_languagetool.py` esegue questa verifica e genera un report Markdown dettagliato segnalando ogni errore rilevante con il numero della regola violata, il tipo di errore, il messaggio esplicativo e i suggerimenti di correzione.

5.1.3 Controllo Sintattico \LaTeX (ChkTeX)

ChkTeX è un linter specifico per file \LaTeX che verifica la correttezza sintattica e stilistica del codice sorgente. Controlla:

- Correttezza dei comandi \LaTeX ;
- Bilanciamento delle parentesi e delle graffe;
- Uso corretto degli spazi prima e dopo i comandi;
- Altre convenzioni stilistiche di buona pratica \LaTeX .

Lo script `check_chktex.py` esegue ChkTeX su tutti i file `.tex` e filtra alcuni warning innocui (ad esempio riferimenti a comandi personalizzati o numeri di versione non seguiti da unità). Il report viene salvato in formato JSON.

5.2 Soglie di Accettazione e Target

Le seguenti tabelle definiscono i target di qualità e le soglie di accettazione per le metriche documentali:

Metrica	Descrizione	Ottimale	Accettabile
Gulpease	Indice di leggibilità per testi in italiano	≥ 60	≥ 45

Tabella 6: Soglie per l'indice Gulpease

Indicatore	Descrizione	Target (PB)
Errori LanguageTool	Numero massimo di segnalazioni rilevanti per file	≤ 10
Errori ChkTeX	Numero massimo di warning significativi per file	≤ 5

Tabella 7: Soglie per gli errori linguistici e \LaTeX

5.3 Grafici

5.3.1 Indice Gulpease

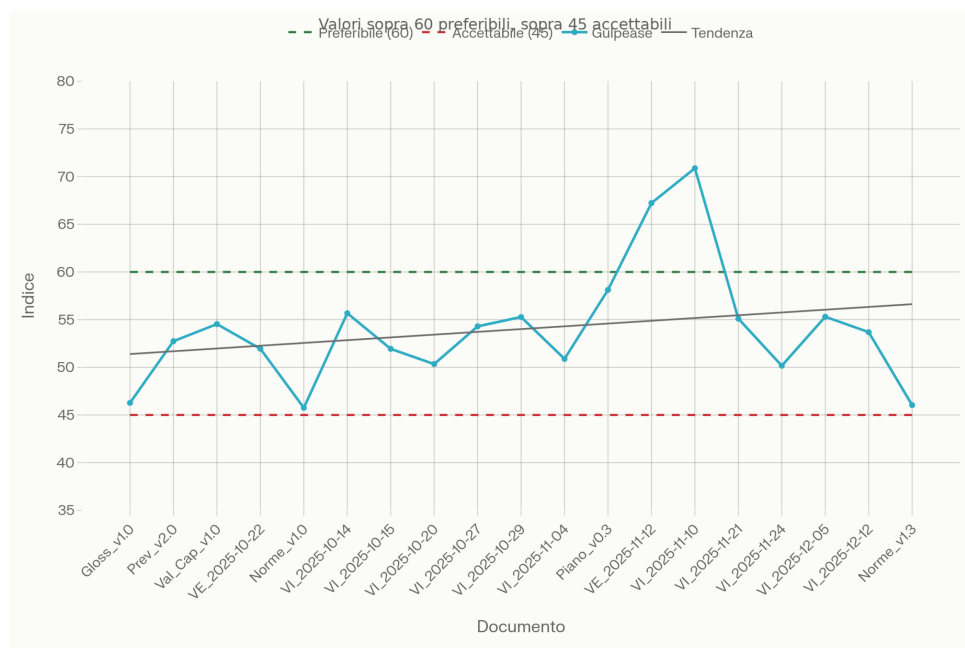


Figura 1: Andamento indice Gulpease

5.3.2 LanguageTool

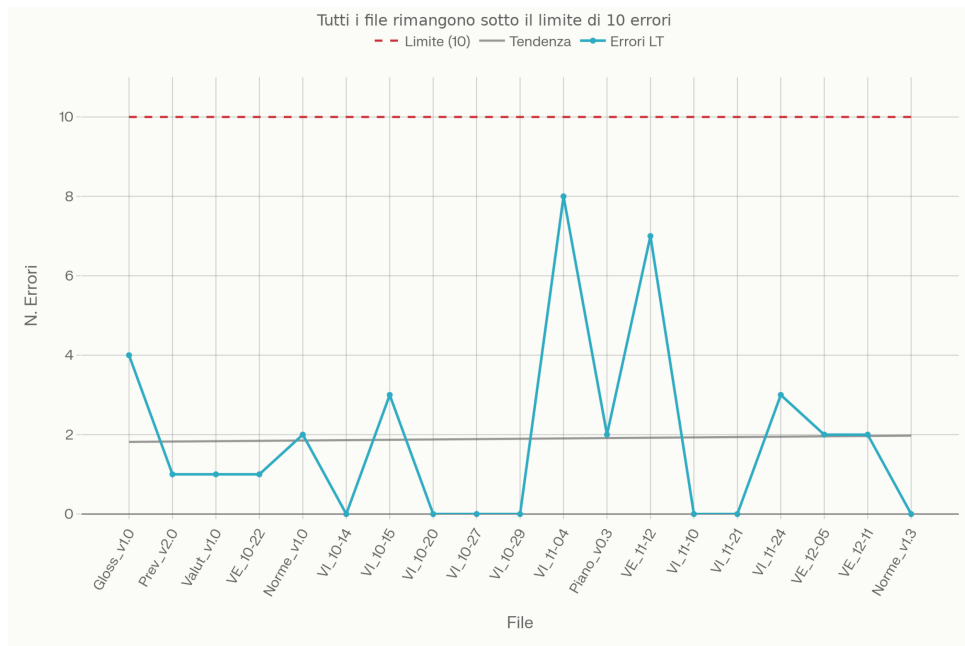


Figura 2: Andamento errori LanguageTool

5.3.3 ChkTeX

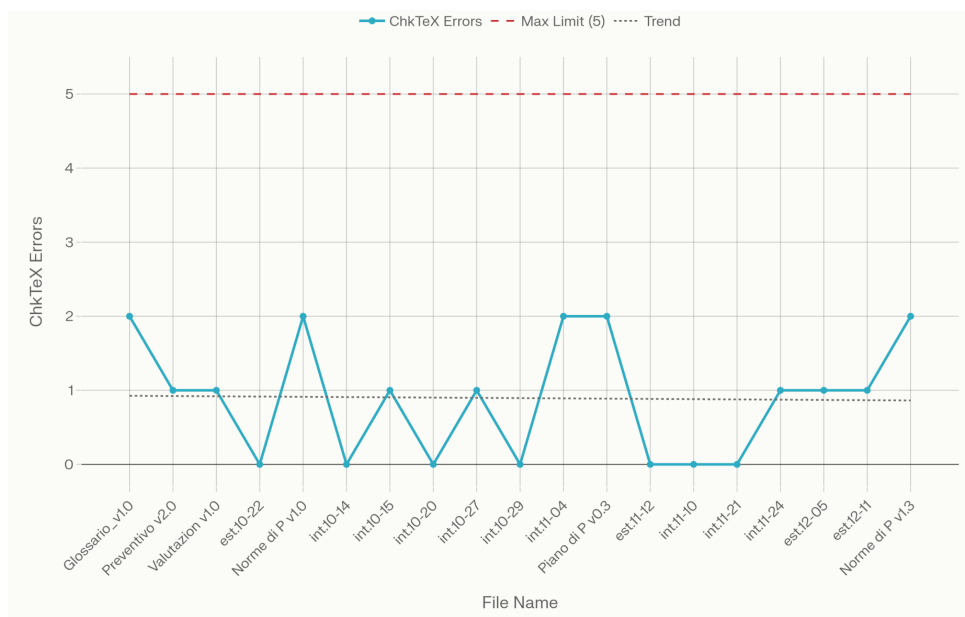


Figura 3: Andamento errori ChkTeX

5.4 Automazione e Verifica Continua

Le verifiche di qualità della documentazione sono automatizzate e integrate nel processo di CI/CD tramite il workflow GitHub `quality_checks.yml`. Questo workflow:

1. Si attiva automaticamente dopo il completamento della compilazione $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ (workflow *Build LaTeX documents*);

2. Può essere eseguito manualmente in qualsiasi momento tramite *workflow dispatch*;
3. Esegue in sequenza i tre script di verifica (Gulpease, LanguageTool, ChkTeX);
4. Genera un report Markdown consolidato accessibile come artifact;
5. Commenta automaticamente i pull request con i risultati della verifica.

La verifica è non-bloccante per i build, ovvero gli errori rilevati non impediscono il progresso della pipeline, tuttavia vengono tracciati e riportati nel report per consentire al gruppo di monitorare gli standard di qualità e di correggere sistematicamente i problemi identificati.