



# Piano di Qualifica

NearYou  
Smart custom advertising platform

[sevenbits.swe.unipd@gmail.com](mailto:sevenbits.swe.unipd@gmail.com)



### Registro modifiche

Versione	Data	Autore	Verificatore	Descrizione
2.0.0	2025-04-11	Leonardo Trolese	Manuel Gusella	Approvazione documento per $PB_G$
1.2.1	2025-04-10	Leonardo Trolese	Manuel Gusella	Aggiunta e analisi della restante parte dei grafici aggiornati
1.2.0	2025-04-09	Federico Pivetta	Manuel Gusella	Aggiunta sezione per le metriche con sola misurazione finale
1.1.1	2025-04-08	Leonardo Trolese	Manuel Gusella	Aggiunta e analisi di parte dei grafici aggiornati
1.1.0	2025-03-29	Federico Pivetta	Riccardo Piva	Aggiunta sezione per le metriche escluse dal cruscotto $_G$
1.0.5	2025-03-28	Leonardo Trolese	Riccardo Piva	Ampliamento del cruscotto $_G$ con le metriche relative alla revisione $PB_G$
1.0.4	2025-03-26	Federico Pivetta	Riccardo Piva	Ampliamento del cruscotto $_G$ con le metriche relative alla revisione $PB_G$
1.0.3	2025-03-26	Alfredo Rubino	Riccardo Piva	Redazione sottosezione Test $_G$ di Unità
1.0.2	2025-03-25	Leonardo Trolese	Riccardo Piva	Aggiornamento metriche a seguito di inizio stesura codice
1.0.1	2025-03-21	Federico Pivetta	Uncas Peruzzi	Correzione ai test $_G$ di sistema $_G$
1.0.0	2025-02-21	Giovanni Cristellon	Leonardo Trolese	Approvazione documento per RTB $_G$
0.4.8	2025-02-21	Leonardo Trolese	Manuel Gusella	Correzione errori minori
0.4.7	2025-02-18	Leonardo Trolese	Pivetta Federico	Aggiunta ulteriori grafici sulle metriche del cruscotto di valutazione
0.4.6	2025-02-16	Leonardo Trolese	Pivetta Federico	Aggiunta ulteriori grafici sulle metriche del cruscotto di valutazione
0.4.5	2025-02-15	Leonardo Trolese	Pivetta Federico	Aggiunta grafici sulle metriche del cruscotto di valutazione
0.4.4	2025-02-11	Federico Pivetta	Uncas Peruzzi	Correzione ai test $_G$ di sistema $_G$ e di accettazione a seguito delle modifiche all'Analisi dei Requisiti
0.4.3	2025-01-14	Federico Pivetta	Leonardo Trolese	Riorganizzazione delle metriche di qualità
0.4.2	2025-01-10	Federico Pivetta	Leonardo Trolese	Aggiunta dei test $_G$ di sistema $_G$ e dei test $_G$ di accettazione
0.4.1	2025-01-08	Riccardo Piva	Uncas Peruzzi	Refactor generale sezione qualità processo $_G$ e qualità prodotto

Versione	Data	Autore	Verificatore	Descrizione
0.4.0	2025-01-07	Riccardo Piva	Uncas Peruzzi	Creazione cruscotto <sub>G</sub>
0.3.3	2025-01-03	Riccardo Piva	Uncas Peruzzi	Correzioni minori generali
0.3.2	2024-12-16	Alfredo Rubino	Manuel Gusella	Aggiunta acronimi metriche e correzioni minori
0.3.1	2024-12-13	Riccardo Piva	Alfredo Rubino	Correzione standard IEEE <sub>G</sub>
0.3.0	2024-12-12	Riccardo Piva	Alfredo Rubino	Arricchimento sezioni Qualità di processo <sub>G</sub> , Qualità di prodotto e inizio redazione modalità testing
0.2.0	2024-12-06	Manuel Gusella	Alfredo Rubino	Inizio redazione sottosezione Qualità di prodotto
0.1.0	2024-11-21	Uncas Peruzzi	Federico Pivetta	Inizio redazione del documento

# Indice

<b>1</b>	<b>Introduzione</b>	<b>7</b>
1.1	Scopo del documento . . . . .	7
1.2	Glossario . . . . .	7
1.3	Riferimenti . . . . .	7
1.3.1	Riferimenti normativi . . . . .	7
1.3.2	Riferimenti informativi . . . . .	7
<b>2</b>	<b>Obiettivi metrici di qualità</b>	<b>8</b>
2.1	Qualità di prodotto . . . . .	8
2.1.1	Funzionalità . . . . .	8
2.1.2	Affidabilità . . . . .	8
2.1.3	Efficienza . . . . .	8
2.1.4	Usabilità . . . . .	8
2.1.5	Manutenibilità . . . . .	9
2.1.6	Portabilità . . . . .	9
2.2	Qualità di processo . . . . .	9
2.2.1	Processi Primari . . . . .	9
2.2.1.1	Fornitura . . . . .	9
2.2.1.2	Sviluppo . . . . .	10
2.2.2	Processi di Supporto . . . . .	10
2.2.2.1	Documentazione . . . . .	10
2.2.2.2	Verifica . . . . .	10
2.2.2.3	Gestione della qualità . . . . .	10
2.2.3	Processi Organizzativi . . . . .	10
2.2.3.1	Gestione dei processi . . . . .	11
<b>3</b>	<b>Modalità di Testing</b>	<b>12</b>
3.1	Test di unità . . . . .	12
3.2	Test di sistema . . . . .	16
3.3	Test di integrazione . . . . .	19
3.4	Test di accettazione . . . . .	19
<b>4</b>	<b>Cruscotto di valutazione delle qualità</b>	<b>21</b>
4.1	Qualità di processo - Fornitura . . . . .	21
4.1.1	MPC01 - Estimated at completion (EAC) . . . . .	21
4.1.2	MPC05 - Planned Value (PV) & MPC04 - Earned Value (EV) . . . . .	22
4.1.3	MPC03 - Actual Cost (AC) & MPC02 - Estimate to Complete (ETC) . . . . .	23
4.1.4	MPC08 - Cost Performance Index (CPI) . . . . .	24
4.1.5	MPC07 - Cost Variance (CV) & MPC06 - Schedule Variance (SV) . . . . .	25
4.2	Qualità di processo - Sviluppo . . . . .	26
4.2.1	MPC10 - Requirements Stability Index (RSI) . . . . .	26
4.3	Qualità di processo - Gestione dei processi . . . . .	27
4.3.1	MPC16 - Rischi non previsti . . . . .	27
4.3.2	MPC17 - Efficienza temporale (ET) . . . . .	27
4.4	Qualità di processo - Documentazione . . . . .	28
4.4.1	MPC01 - Errori ortografici . . . . .	28
4.4.2	MPC11 - Indice Gulpease . . . . .	29
4.5	Qualità di processo - Gestione della qualità . . . . .	30
4.5.1	MPC15 - Metriche di qualità soddisfatte . . . . .	30
4.6	Qualità di processo - Analisi dei requisiti . . . . .	31
4.6.1	MPC09 - Requisiti Obbligatori Soddisfatti (ROS) . . . . .	31
4.7	Qualità di processo - Verifica . . . . .	32
4.7.1	MPD04 - Line Coverage . . . . .	32
4.7.2	MPD05 - Branch Coverage . . . . .	33
4.7.3	MPD09 - Linee di codice per metodo . . . . .	34
4.7.4	MPD10 - Attributi per classe . . . . .	35
4.7.5	MPD11 - Structure Fan IN . . . . .	36

---

4.7.6	MPD12 - Structure Fan OUT . . . . .	37
4.7.7	MPC14 - Passed test cases percentage . . . . .	38
<b>5</b>	<b>Metriche non incluse nel cruscotto</b>	<b>38</b>

## Elenco delle figure

1	Grafico a linee della metrica EAC . . . . .	21
2	Grafico a linee delle metriche EV e PV . . . . .	22
3	Grafico a linee delle metriche AC e ETC . . . . .	23
4	Grafico a linee della metrica CPI . . . . .	24
5	Grafico a linee delle metriche CV e SV . . . . .	25
6	Grafico a linee della metrica RSI . . . . .	26
7	Grafico a linee della metrica "Rischi non previsti" . . . . .	27
8	Grafico a linee della metrica ET . . . . .	27
9	Grafico a linee della metrica "Errori ortografici" . . . . .	28
10	Grafico a linee della metrica "Indice di Gulpease" . . . . .	29
11	Grafico a linee della metrica "metriche di qualità soddisfatte" . . . . .	30
12	Grafico a linee della metrica ROS . . . . .	31
13	Grafico a linee della metrica "Line coverage" . . . . .	32
14	Grafico a linee della metrica "Branch coverage" . . . . .	33
15	Grafico a barre della metrica "Linee di codice per metodo" . . . . .	34
16	Grafico a barre della metrica "Attributi per classe" . . . . .	35
17	Grafico a barre della metrica "Structure Fan IN" . . . . .	36
18	Grafico a barre della metrica "Structure Fan OUT" . . . . .	37
19	Grafico a linee della metrica "Passed test cases percentage" . . . . .	38

## Elenco delle tabelle

2	Funzionalità - Qualità di prodotto . . . . .	8
3	Affidabilità - Qualità di prodotto . . . . .	8
4	Efficienza - Qualità di prodotto . . . . .	8
5	Usabilità - Qualità di prodotto . . . . .	8
6	Manutenibilità - Qualità di prodotto . . . . .	9
7	Portabilità - Qualità di prodotto . . . . .	9
8	Processi primari - Fornitura . . . . .	9
9	Processi primari - Codifica . . . . .	10
10	Processi di supporto - Documentazione . . . . .	10
11	Processi di supporto - Verifica . . . . .	10
12	Processi di supporto - Gestione della qualità . . . . .	10
13	Processi organizzativi - Gestione dei processi . . . . .	11
14	Test di unità . . . . .	16
15	Test di sistema . . . . .	19
16	Test di integrazione . . . . .	19
17	Test di accettazione . . . . .	21
18	Metriche di qualità non incluse nel cruscotto . . . . .	38

# 1 Introduzione

## 1.1 Scopo del documento

Il seguente documento ha l'obiettivo di garantire la qualità del prodotto e dei processi coinvolti nell'intero progetto<sub>G</sub>. Al fine di assicurare che il prodotto soddisfi le aspettative di qualità attese, il documento verrà aggiornato nel tempo per riflettere eventuali modifiche, integrazioni e i risultati delle verifiche effettuate.

## 1.2 Glossario

Con l'intento di evitare ambiguità nell'interpretazione del linguaggio utilizzato, viene fornito un glossario che si occupa di esplicitare il significato dei termini che riguardano il contesto del progetto<sub>G</sub>. I termini presenti nel glossario sono contrassegnati con una <sub>G</sub> a pedice : Termine<sub>G</sub>.

Le definizioni sono contenute nell'apposito documento *Glossario*.

## 1.3 Riferimenti

### 1.3.1 Riferimenti normativi

- Norme di Progetto\_v1.0.0
- Regolamento del progetto<sub>G</sub> didattico  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/PD1.pdf>  
(Consultato: 2025-02-19).
- Capitolato<sub>G</sub> C4 - NearYou - Smart custom advertising platform  
<https://www.math.unipd.it/~tullio/IS-1/2024/Progetto/C4p.pdf>  
(Consultato: 2025-02-19).  
<https://www.math.unipd.it/~tullio/IS-1/2024/Progetto/C4.pdf>  
(Consultato: 2025-02-19).
- Standard ISO/IEC 9126  
[https://en.wikipedia.org/wiki/ISO/IEC\\_9126](https://en.wikipedia.org/wiki/ISO/IEC_9126)  
(Consultato: 2025-02-19).
- Standard ISO/IEC/IEEE 12207:1995  
[https://www.math.unipd.it/~tullio/IS-1/2009/Approfondimenti/ISO\\_12207-1995.pdf](https://www.math.unipd.it/~tullio/IS-1/2009/Approfondimenti/ISO_12207-1995.pdf)  
(Consultato: 2025-02-19).

### 1.3.2 Riferimenti informativi

- Qualità di prodotto  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T07.pdf>  
(Consultato: 2025-02-19).
- Qualità di processo<sub>G</sub>  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T08.pdf>  
(Consultato: 2025-02-19).
- Verifica e validazione  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T09.pdf>  
(Consultato: 2025-02-19).  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T10.pdf>  
(Consultato: 2025-02-19).  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T11.pdf>  
(Consultato: 2025-02-19).



## 2 Obiettivi metrici di qualità

Per far sì che un prodotto raggiunga uno standard qualitativo, è necessario definire delle metriche precise che permettano di monitorare e indicare il grado di qualità del prodotto e che quindi permettano di definire questo standard. Queste metriche vengono definite nel documento *Norme di Progetto<sub>G</sub>*. Questa sezione si occuperà di definire i parametri di accettazione e ottimalità delle relative metriche.

### 2.1 Qualità di prodotto

La qualità di prodotto è intesa come valutazione del software, e più precisamente per la determinazione del grado di conformità alle attese.

Si rivolge l'attenzione su aspetti come Usabilità, Affidabilità e Manutenibilità, ma più in generale alla qualità esterna (funzionale) ed interna (strutturale) del prodotto software.

Quindi non basta che il software implementi le funzionalità richieste dal proponente<sub>G</sub>, ma le esegua secondo specifici standard di qualità.

In seguito sono presenti le metriche definite dallo standard ISO/IEC 9126 che il gruppo si impegna a soddisfare per la qualità del prodotto software.

#### 2.1.1 Funzionalità

Metrica	Descrizione	Valore accettazione	Valore ideale
MPD01	Requisiti Obbligatori Soddisfatti	100%	100%
MPD02	Requisiti Desiderabili Soddisfatti	$\geq 0\%$	100%
MPD03	Requisiti Opzionali Soddisfatti	$\geq 0\%$	100%

Table 2: Funzionalità - Qualità di prodotto

#### 2.1.2 Affidabilità

Metrica	Descrizione	Valore accettazione	Valore ideale
MPD04	Line coverage	$\geq 80\%$	100%
MPD05	Branch <sub>G</sub> coverage	$\geq 80\%$	100%

Table 3: Affidabilità - Qualità di prodotto

#### 2.1.3 Efficienza

Metrica	Descrizione	Valore accettazione	Valore ideale
MPD06	Tempo medio di risposta	$\leq 10$ secondi	$\leq 4$ secondi

Table 4: Efficienza - Qualità di prodotto

#### 2.1.4 Usabilità

Metrica	Descrizione	Valore accettazione	Valore ideale
MPD07	Facilità di utilizzo	$\leq 7$ click	$\leq 5$ click
MPD08	Tempo medio di apprendimento	$\leq 5$ minuti	$\leq 2$ minuti

Table 5: Usabilità - Qualità di prodotto

### 2.1.5 Manutenibilità

Metrica	Descrizione	Valore accettazione	Valore ideale
MPD09	Linee di codice per metodo	$\leq 50$	$\leq 25$
MPD10	Attributi per classe	$\leq 7$	$\leq 5$
MPD11	Structure Fan IN	-	-
MPD12	Structure Fan OUT	-	va minimizzato

Table 6: Manutenibilità - Qualità di prodotto

### 2.1.6 Portabilità

Metrica	Descrizione	Valore accettazione	Valore ideale
MPD13	Versioni browser supportati	$\geq 80\%$	100%

Table 7: Portabilità - Qualità di prodotto

## 2.2 Qualità di processo

La qualità di processo<sub>G</sub> è intesa come valutazione delle attività svolte per la realizzazione del prodotto. Seguendo delle buone pratiche e delle linee guida nello sviluppo software, si può garantire che il prodotto finale avrà rispettato a sua volta degli standard qualitativi rendendolo così un prodotto di qualità. Qui sotto divideremo le metriche di qualità di processo<sub>G</sub> seguendo lo standard ISO/IEC 12207:1995 in tre categorie: Processi primari, Processi di supporto e Processi organizzativi.

### 2.2.1 Processi Primari

I processi primari si possono dividere in parti primarie e una parte primaria è quella che inizia o esegue lo sviluppo, l'operazione o la manutenzione di prodotti software.

#### 2.2.1.1 Fornitura

La fornitura è il processo<sub>G</sub> che si occupa di consegnare il prodotto software al cliente. Serve per garantire che il prodotto soddisfi i requisiti di tempi e costi definiti con il cliente.

Metrica	Descrizione	Valore accettazione	Valore ideale
MPC01	Estimated at completion (EAC)	$\pm 5\%$ rispetto al $(BAC)_G$	Budget at completion $(BAC)_G$
MPC02	Estimate to complete (ETC)	$\geq 0$	$\leq EAC_{GG}$
MPC03	Actual cost (AC)	$\geq 0$	$\leq EAC_{GG}$
MPC04	Earned value (EV)	$\geq 0$	$\leq EAC_{GG}$
MPC05	Planned value (PV)	$\geq 0$	$\leq$ Budget at completion $(BAC)_G$
MPC06	Schedule variance (SV)	$\geq -5\%$ rispetto al $(BAC)_G$	$\geq 0\%$
MPC07	Cost variance (CV)	$\geq -5\%$ rispetto al $(BAC)_G$	$\geq 0\%$
MPC08	Cost Performance Index (CPI)	$\geq 0.9$	$\geq 1.0$

Table 8: Processi primari - Fornitura

### 2.2.1.2 Sviluppo

Lo sviluppo è il processo<sub>G</sub> riguardante la scrittura del codice del prodotto software.

Questa metrica serve a garantire che il software rispetti le richieste del cliente e che la codifica avvenga in modo efficiente

Metrica	Descrizione	Valore accettazione	Valore ideale
MPC09	Requisiti Obbligatori Soddisfatti (ROS)	100%	100%
MPC10	Requirements Stability Index (RSI)	$\geq 80\%$	100%

Table 9: Processi primari - Codifica

## 2.2.2 Processi di Supporto

Un processo<sub>G</sub> di supporto è un processo<sub>G</sub> che supporta un altro processo<sub>G</sub> come parte integrante con uno scopo distinto e contribuisce al successo e alla qualità del progetto<sub>G</sub> software.

Un processo<sub>G</sub> di supporto è impiegato ed eseguito, se necessario, da un altro processo<sub>G</sub>.

### 2.2.2.1 Documentazione

La documentazione<sub>G</sub> è essenziale per la comprensione del prodotto e per la sua manutenzione.

Di conseguenza è essenziale che questa sia chiara, comprensibile e corretta.

Metrica	Descrizione	Valore accettazione	Valore ideale
MPC11	Indice Gulpease	$\geq 40\%$	$\geq 60\%$
MPC12	Correttezza ortografica	0 errori	0 errori

Table 10: Processi di supporto - Documentazione

### 2.2.2.2 Verifica

La verifica serve a garantire che il prodotto software sia conforme alle specifiche e non contenga errori.

Metrica	Descrizione	Valore accettazione	Valore ideale
MPC13	Line coverage	$\geq 80\%$	100%
MPC14	Passed test <sub>G</sub> cases percentage	$\geq 80\%$	100%

Table 11: Processi di supporto - Verifica

### 2.2.2.3 Gestione della qualità

La gestione della qualità è necessaria per garantire che tutte le metriche di qualità vengano effettivamente soddisfatte.

Metrica	Descrizione	Valore accettazione	Valore ideale
MPC15	Metriche di qualità soddisfatte	$\geq 85\%$	100%

Table 12: Processi di supporto - Gestione della qualità

## 2.2.3 Processi Organizzativi

I processi organizzativi servono per creare un sottostruttura per il ciclo-di-vita<sub>G</sub> e per garantire che i processi principali e i loro processi di supporto siano ben strutturati e vengano continuamente migliorati.

### 2.2.3.1 Gestione dei processi

La gestione dei processi indica come vengono gestiti i processi all'interno del progetto<sub>G</sub>.

Metrica	Descrizione	Valore accettazione	Valore ideale
MPC16	Rischi non previsti	$\leq 3$	0
MPC17	Efficienza <sub>G</sub> temporale (ET)	$\leq 5.0$	$\geq 1.0$

Table 13: Processi organizzativi - Gestione dei processi

### 3 Modalità di Testing

Qui sotto sono elencati i vari  $\text{test}_G$  che vengono eseguiti automaticamente sul prodotto software. Questo serve a garantire che il prodotto soddisfi i requisiti e le aspettative indicate nel documento *Analisi dei Requisiti*.

I  $\text{test}_G$  sono divisi in quattro categorie:  $\text{Test}_G$  di unità,  $\text{Test}_G$  di sistema $_G$ ,  $\text{Test}_G$  di integrazione e  $\text{Test}_G$  di accettazione.

Per indicarne lo stato, come definito nel documento *Norme di Progetto $_G$*  vengono utilizzate le seguenti abbreviazioni:

- **P**: Passato
- **NP**: Non Passato
- **NI**: Non Implementato

#### 3.1 Test di unità

I  $\text{test}_G$  di unità servono a verificare che ogni singola unità del software funzioni correttamente.

Codice	Descrizione	Stato
TU1	Verificare che la classe <code>ClickhouseActivityRepository</code> restituisca correttamente le attività entro un determinato raggio geografico e i dettagli di una specifica attività a partire dal nome. Se l'attività non viene trovata, deve restituire un oggetto vuoto.	P
TU2	Verificare che la classe <code>ClickhouseMessageRepository</code> restituisca correttamente l'ultimo messaggio inviato da un utente, includendo tutte le informazioni associate. In caso di assenza di messaggi, deve restituire un oggetto vuoto con valori di default.	P
TU3	Verificare che la classe <code>ClickhouseUserRepository</code> gestisca correttamente l'assegnazione e il recupero delle informazioni sugli utenti. In particolare, deve aggiornare lo stato di un utente come occupato, recuperare un utente libero, restituire i dettagli di un utente associato a un sensore e gestire correttamente i casi in cui nessun utente venga trovato.	P
TU4	Verificare che la classe <code>CustomPrompt</code> generi correttamente un prompt formattato in base alle informazioni dell'utente e alle attività disponibili. Deve garantire che i dati forniti siano integrati correttamente nel testo generato e che venga mantenuta una struttura valida anche in caso di dati mancanti.	P

Codice	Descrizione	Stato
TU5	Verificare che la classe <b>DatabaseConnection</b> gestisca correttamente la connessione e la disconnessione al database <sub>G</sub> . Inoltre, si verifica che la classe <b>DatabaseConfigParameters</b> inizializzi correttamente i valori di default dei parametri di configurazione.	P
TU6	Verificare che le classi DTO ( <b>MessageDTO</b> , <b>UserDTO</b> , <b>ActivityDTO</b> ) inizializzino correttamente i dati forniti e gestiscano i valori di default quando i parametri non vengono specificati.	P
TU7	Verificare che la classe <b>FilterMessageAlreadyDisplayed</b> filtri correttamente i messaggi già visualizzati dall'utente, escludendo i messaggi con le stesse coordinate dell'ultimo messaggio inviato, i messaggi con coordinate (0,0) e quelli che risultano identici dopo l'arrotondamento delle coordinate.	P
TU8	Verificare che la classe <b>FlinkJobManager</b> inizializzi e gestisca correttamente il flusso di dati in Apache Flink <sub>G</sub> . Deve configurare correttamente le operazioni di <b>map</b> e <b>filter</b> , collegarsi a sorgenti e sink e avviare l'esecuzione del job tramite <b>execute</b> .	P
TU9	Verificare che la classe <b>GroqLLMService</b> gestisca correttamente l'inizializzazione, la configurazione della chat e la generazione di risposte strutturate. In particolare, deve caricare la chiave API <sub>G</sub> dall'ambiente, impostare correttamente il modello <b>ChatGroq</b> con i parametri previsti e restituire un output coerente con la richiesta ricevuta.	P
TU10	Verificare che la classe <b>JsonRowDeserializationAdapter</b> inizializzi correttamente lo schema di deserializzazione JSON <sub>G</sub> , memorizzando correttamente il tipo di dati fornito e creando un'istanza di <b>JsonRowDeserializationSchema</b> con i parametri appropriati.	P
TU11	Verificare che la classe <b>JsonRowSerializationAdapter</b> inizializzi correttamente lo schema di serializzazione JSON <sub>G</sub> . In particolare, deve memorizzare il tipo di dati fornito e creare un'istanza di <b>JsonRowSerializationSchema</b> configurata con i parametri appropriati.	P

Codice	Descrizione	Stato
TU12	Verificare che la classe <code>KafkaMessageWriter</code> inizializzi correttamente il writer <code>Kafka<sub>G</sub></code> . In particolare, deve configurare lo schema di serializzazione, impostare i parametri del sink <code>Kafka<sub>G</sub></code> e restituire correttamente l'istanza del writer tramite il metodo <code>get_message_writer</code> .	P
TU13	Verificare che la classe <code>KafkaPositionReceiver</code> configuri correttamente la sorgente <code>Kafka<sub>G</sub></code> . In particolare, deve impostare i parametri di connessione, il gruppo di consumo, il deserializzatore e le proprietà di gestione degli offset, restituendo correttamente l'istanza della sorgente tramite il metodo <code>get_position_receiver</code> .	P
TU14	Verificare che la classe <code>KafkaSourceConfiguration</code> inizializzi correttamente i parametri di configurazione con i valori di default previsti, inclusi server, <code>topic<sub>G</sub></code> , gruppo di consumo e gestione degli offset.	P
TU15	Verificare che la classe <code>KafkaWriterConfiguration</code> inizializzi correttamente i parametri di configurazione con i valori di default previsti. In particolare, deve garantire che il server, il <code>topic<sub>G</sub></code> scrivibile e la struttura del <code>key_type</code> siano definiti correttamente.	P
TU16	Verificare che la classe <code>MessageSerializer</code> crei correttamente un oggetto <code>Row</code> a partire da un'istanza di <code>MessageDTO</code> . In particolare, deve garantire che i dati siano convertiti correttamente e mantenere la struttura prevista con il numero corretto di campi.	P
TU17	Verificare che la classe <code>PositionToMessageProcessor</code> elabori correttamente le posizioni degli utenti e generi messaggi pubblicitari basati sulle attività nelle vicinanze. In particolare, deve gestire i casi in cui non ci siano attività disponibili e interagire con i repository <code>G</code> e il servizio AI per elaborare i dati.	P
TU18	Verificare che la classe <code>BycicleSimulationStrategy</code> inizializzi correttamente i parametri di simulazione e calcoli correttamente il percorso <code>G</code> , la velocità e il tempo tra le posizioni. In particolare, deve interagire con il grafo per ottenere un percorso <code>G</code> valido e restituire valori coerenti per velocità e intervallo di tempo.	P

Codice	Descrizione	Stato
TU19	Verificare che la classe <b>DatabaseConfigParameters</b> inizializzi correttamente i parametri di configurazione con i valori di default previsti.	P
TU20	Verificare che la classe <b>DatabaseConnection</b> gestisca correttamente la connessione e la disconnessione dal database <sub>G</sub> .	P
TU21	Verificare che la classe <b>GeoPosition</b> inizializzi correttamente i parametri e restituisca i valori attesi tramite i metodi di accesso. In particolare, deve garantire che <b>sensor_id</b> , latitudine, longitudine e timestamp siano correttamente memorizzati e recuperabili.	P
TU22	Verificare che la classe <b>GpsSensor</b> gestisca correttamente la simulazione della posizione. In particolare, deve inizializzare correttamente i parametri, generare e inviare posizioni simulate basate sulla strategia di simulazione e restituire i valori attesi per identificativo del sensore e tempo di aggiornamento.	P
TU23	Verificare che la classe <b>GraphWrapper</b> inizializzi e recuperi correttamente la mappa stradale utilizzando <b>osmnx</b> . In particolare, deve chiamare <b>graph_from_point</b> con i parametri corretti per ottenere il grafo stradale basato sulla posizione, il raggio e il tipo di rete specificati.	P
TU24	Verificare che la classe <b>KafkaConfigParameters</b> inizializzi correttamente i parametri di configurazione con i valori di default previsti, inclusi il server Kafka <sub>G</sub> e il topic <sub>G</sub> di origine.	P
TU25	Verificare che la classe <b>KafkaConfluentAdapter</b> invii correttamente i dati al broker <sub>G</sub> Kafka <sub>G</sub> . In particolare, deve serializzare correttamente le posizioni, produrre i messaggi con la chiave corretta e gestire la sincronizzazione dei thread tramite un meccanismo di lock.	P
TU26	Verificare che la classe <b>PositionJsonAdapter</b> serializzi correttamente un oggetto <b>GeoPosition</b> in formato JSON <sub>G</sub> , mantenendo la struttura e i valori attesi per identificativo, coordinate e timestamp.	P



Codice	Descrizione	Stato
TU27	Verificare che la classe <code>SensorFactory</code> crei correttamente istanze di <code>GpsSensor</code> . In particolare, deve garantire che ogni sensore sia un'istanza valida di <code>SensorSubject</code> e <code>GpsSensor</code> , che gli UUID assegnati siano validi e univoci, e che il metodo per la creazione di una lista di sensori generi il numero corretto di istanze.	P
TU28	Verificare che la classe <code>SensorRepository</code> gestisca correttamente l'assegnazione e il recupero dei sensori. In particolare, deve aggiornare lo stato di un sensore come occupato e recuperare un sensore non occupato dal database <sub>G</sub> .	P
TU29	Verificare che la classe <code>SensorSimulationAdministrator</code> gestisca correttamente l'avvio della simulazione dei sensori. In particolare, deve inizializzare correttamente il registro dei sensori, avviare la simulazione tramite un <code>ThreadPool</code> e gestire eventuali errori durante l'esecuzione.	P
TU30	Verificare che la classe <code>UserRepository</code> gestisca correttamente l'assegnazione e il recupero degli utenti. In particolare, deve aggiornare lo stato di un utente come occupato e recuperare un utente libero dal database <sub>G</sub> .	P
TU31	Verificare che la classe <code>UserSensorService</code> assegni correttamente un sensore a un utente disponibile. In particolare, deve recuperare un utente e un sensore non occupati, aggiornare il loro stato nel database <sub>G</sub> e gestire correttamente i casi in cui non siano disponibili utenti o sensori.	P

Table 14: Test di unità

### 3.2 Test di sistema

I test<sub>G</sub> di sistema<sub>G</sub> servono a verificare la completa copertura dei requisiti concordati nel documento *Analisi dei Requisiti*.

Codice	Descrizione	Requisito	Stato
TS1	Verificare che l'utente privilegiato possa visualizzare la Dashboard <sub>G</sub> composta da una mappa interattiva con i vari Marker <sub>G</sub> su di essa.	RF01	P
TS2	Verificare che l'utente privilegiato possa visualizzare dei Marker <sub>G</sub> che rappresentano i vari Percorsi <sub>G</sub> effettuati in tempo reale dagli utenti presenti nel Sistema <sub>G</sub>	RF02	P

Codice	Descrizione	Requisito	Stato
TS3	Verificare che l'utente privilegiato possa visualizzare un Marker <sub>G</sub> che rappresenta un Percorso <sub>G</sub> effettuato in tempo reale da un utente presente nel Sistema <sub>G</sub> .	RF03	P
TS4	Verificare che l'utente privilegiato possa visualizzare tutti i punti di interesse riconosciuti dal Sistema <sub>G</sub> .	RF04	P
TS5	Verificare che l'utente privilegiato possa visualizzare un Marker <sub>G</sub> che rappresenta un punto di interesse riconosciuto dal Sistema <sub>G</sub> .	RF05	P
TS6	Verificare che l'utente privilegiato possa visualizzare gli annunci pubblicitari provenienti da un determinato punto di interesse.	RF06	P
TS7	Verificare che l'utente privilegiato possa visualizzare un singolo annuncio pubblicitario tramite un Marker <sub>G</sub> .	RF07	P
TS8	Verificare che l'utente privilegiato possa visualizzare una Dashboard <sub>G</sub> relativa ad un singolo utente quando seleziona un Marker <sub>G</sub> utente nella Dashboard <sub>G</sub> principale.	RF08	P
TS9	Verificare che l'utente privilegiato possa visualizzare dei Marker <sub>G</sub> che rappresentano lo storico delle posizioni dell'utente a cui è riferita la Dashboard <sub>G</sub> di singolo utente.	RF09	P
TS10	Verificare che l'utente privilegiato possa visualizzare un Marker <sub>G</sub> che rappresenta la posizione dell'utente in un determinato istante nella Dashboard <sub>G</sub> di singolo utente.	RF10	P
TS11	Verificare che l'utente privilegiato possa visualizzare, nella Dashboard <sub>G</sub> di singolo utente, tutti i punti di interesse riconosciuti dal Sistema <sub>G</sub> .	RF11	P
TS12	Verificare che l'utente privilegiato possa visualizzare, nella Dashboard <sub>G</sub> di singolo utente, un Marker <sub>G</sub> che rappresenta un punto di interesse riconosciuto dal Sistema <sub>G</sub> .	RF12	P
TS13	Verificare che l'utente privilegiato possa visualizzare lo storico degli annunci pubblicitari generati per l'utente a cui è riferita la Dashboard <sub>G</sub> singolo utente.	RF13	P
TS14	Verificare che l'utente privilegiato possa visualizzare un singolo annuncio pubblicitario tramite un Marker <sub>G</sub> nella Dashboard <sub>G</sub> di singolo utente.	RF14	P
TS15	Verificare che l'utente privilegiato possa visualizzare un pannello <sub>G</sub> apposito contenente le informazioni dell'utente, a cui è riferita la Dashboard <sub>G</sub> di singolo utente, in forma tabellare.	RF15	P

Codice	Descrizione	Requisito	Stato
TS16	Verificare che l'utente privilegiato possa visualizzare nel pannello <sub>G</sub> apposito di visualizzazione informazioni dell'utente: il nome, il cognome, l'email, il genere, la data di nascita e lo stato civile.	RF16	P
TS17	Verificare che l'utente privilegiato possa visualizzare i dettagli del Marker <sub>G</sub> riguardante una singola posizione di un utente nella rispettiva Dashboard <sub>G</sub>	RF17	P
TS18	Verificare che l'utente privilegiato quando visualizza i dettagli del Marker <sub>G</sub> , riguardante una singola posizione di un utente nella rispettiva Dashboard <sub>G</sub> , possa vedere la latitudine, la longitudine e l'istante di rilevamento del Marker <sub>G</sub>	RF18	P
TS19	Verificare che l'utente privilegiato possa visualizzare l'area di influenza di un punto di interesse selezionato.	RF19	P
TS20	Verificare che l'utente privilegiato possa visualizzare le informazioni dettagliate di un punto di interesse quando selezionato.	RF20	P
TS21	Verificare che l'utente privilegiato quando visualizza le informazioni dettagliate di un punto di interesse possa visualizzare la latitudine, la longitudine, il nome, la tipologia e la descrizione del punto di interesse.	RF21	P
TS22	Verificare che l'utente possa visualizzare l'annuncio pubblicitario proveniente dal punto di interesse situato nell'area che sta attraversando.	RF22	P
TS23	Verificare che l'utente privilegiato possa visualizzare una tabella contenente le informazioni dei singoli PoI <sub>G</sub> ordinati per la quantità di messaggi inviati nel mese.	RF23	P
TS24	Verificare che l'utente privilegiato possa visualizzare nella tabella dei PoI <sub>G</sub> un singolo PoI <sub>G</sub> , rappresentato da una riga della tabella.	RF24	P
TS25	Verificare che l'utente privilegiato possa visualizzare in ogni riga della tabella dei PoI <sub>G</sub> il nome, l'indirizzo, la tipologia (di che ambito si occupa), la descrizione e il numero di messaggi inviati durante il mese di un singolo PoI <sub>G</sub> .	RF25	P
TS26	Verificare che l'utente privilegiato possa visualizzare i dettagli di un annuncio generato.	RF26	P

Codice	Descrizione	Requisito	Stato
TS27	Verificare che l'utente privilegiato quando visualizza i dettagli di un annuncio possa visualizzare la latitudine, la longitudine, l'istante di creazione, il nome dell'utente coinvolto, il nome del punto di interesse coinvolto e il contenuto dell'annuncio.	RF27	P
TS28	Verificare che il sensore possa trasmettere i dati rilevati in tempo reale al Sistema <sub>G</sub> .	RF28	P
TS29	Verificare che il sensore possa trasmettere il proprio id, la sua latitudine e longitudine al Sistema <sub>G</sub> .	RF29	P

Table 15: Test di sistema

### 3.3 Test di integrazione

I test<sub>G</sub> di integrazione servono a verificare che le componenti del sistema<sub>G</sub> si integrino correttamente e in maniera efficace. L'obiettivo dei test<sub>G</sub> è identificare eventuali problemi di interoperabilità e integrazione fra le componenti del software.

Codice	Descrizione	Stato
TI1	Verificare che i dati simulati vengano correttamente pubblicati sul broker <sub>G</sub> di messaggi	P
TI2	Assicurarsi che il modulo di stream-processing <sub>G</sub> elabori correttamente i dati ricevuti dal broker <sub>G</sub> e li invii al motore di generative AI	P
TI3	Verificare che il motore LLM <sub>G</sub> generi messaggi pubblicitari coerenti e contestualizzati in base ai dati forniti	P
TI4	Controllare che i messaggi generati vengano correttamente salvati nella piattaforma di storage	P
TI5	Controllare che i dati posizionali generati vengano correttamente salvati nella piattaforma di storage	P
TI6	Verificare che i dati geospaziali vengano aggiornati in tempo reale sulla dashboard <sub>G</sub> contenente la mappa	P
TI7	Assicurarsi che i messaggi generati dal motore LLM <sub>G</sub> siano correttamente visualizzati nella dashboard <sub>G</sub>	P

Table 16: Test di integrazione

### 3.4 Test di accettazione

I test<sub>G</sub> di accettazione sono finalizzati a verificare che tutte le esigenze concordate con il proponente<sub>G</sub> siano soddisfatte, e di conseguenza saranno svolti al termine del progetto<sub>G</sub> dai membri del gruppo in

coordinazione con i componenti dell'azienda Synclab<sub>G</sub>.

Codice	Descrizione	Stato
TA1	Verificare che l'utente privilegiato possa visualizzare la Dashboard <sub>G</sub> composta da una mappa e interagire con i Marker <sub>G</sub> presenti su di essa.	P
TA2	Verificare che l'utente privilegiato possa visualizzare sulla mappa tutti i Marker <sub>G</sub> relativi ai Percorsi effettuati in tempo reale dagli utenti, ai punti di interesse riconosciuti dal Sistema <sub>G</sub> e agli annunci pubblicitari associati ai punti di interesse.	P
TA3	Verificare che l'utente privilegiato possa visualizzare una Dashboard <sub>G</sub> relativa ad un singolo utente quando seleziona un Marker <sub>G</sub> utente nella Dashboard <sub>G</sub> principale	P
TA4	Verificare che l'utente privilegiato possa visualizzare, nella Dashboard <sub>G</sub> relativa ad un singolo utente, tutti i punti di interesse riconosciuti dal Sistema <sub>G</sub> , lo storico delle posizioni dell'utente, lo storico degli annunci pubblicitari generati per l'utente, un pannello <sub>G</sub> dedicato con le informazioni dell'utente in forma tabellare e i dettagli relativi a ciascuna posizione.	P
TA5	Verificare che l'utente privilegiato possa visualizzare, nella Dashboard <sub>G</sub> relativa a un singolo utente, i dettagli del pannello <sub>G</sub> , inclusi il nome, il cognome, l'email, il genere, la data di nascita e lo stato civile. Inoltre, verificare che siano visibili i dettagli della singola posizione, inclusi la latitudine, la longitudine e l'istante di rilevamento.	P
TA6	Verificare che l'utente privilegiato possa visualizzare i dettagli di un punto di interesse, inclusi la latitudine, la longitudine, il nome, la tipologia e la descrizione.	P
TA7	Verificare che l'utente privilegiato possa visualizzare i dettagli di un annuncio generato, inclusi la latitudine, la longitudine, l'istante di creazione, il nome dell'utente coinvolto, il nome del punto di interesse coinvolto e il contenuto dell'annuncio.	P
TA8	Verificare che l'utente privilegiato possa visualizzare una tabella contenente l'elenco dei punti di interesse, ordinati per la quantità di messaggi inviati nel mese. Ogni PoI <sub>G</sub> deve essere rappresentato da una riga della tabella, contenente il nome, l'indirizzo, la tipologia, la descrizione e il numero di messaggi inviati.	P

Codice	Descrizione	Stato
TA9	Verificare che il sensore possa trasmettere in tempo reale al Sistema <sub>G</sub> i dati rilevati, inclusi il proprio id, la latitudine e la longitudine.	P

Table 17: Test di accettazione

## 4 Cruscotto di valutazione delle qualità

### 4.1 Qualità di processo - Fornitura

#### 4.1.1 MPC01 - Estimated at completion (EAC)

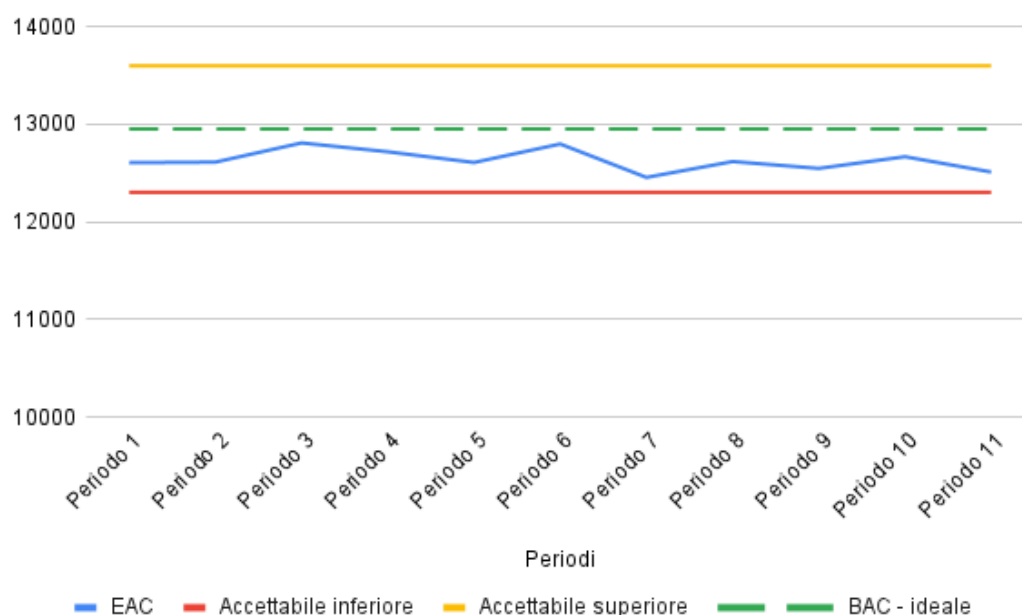


Figure 1: Grafico a linee della metrica EAC

**RTB:** Il grafico rappresenta la stima aggiornata del costo totale del progetto<sub>G</sub> al completamento. Questo valore è quindi determinato dalla somma dei costi sostenuti fino a un certo momento (in termini di ore produttive svolte), e dei costi stimati al completamento (in termini di ore produttive restanti in riferimento al preventivo iniziale del progetto<sub>G</sub>).

In questo caso emerge dal grafico che il valore di EAC<sub>G</sub> è poco al di sotto del preventivo iniziale (BAC), e in ogni caso entro i valori accettabili definiti per la metrica. Questo indica che il progetto<sub>G</sub> è sufficientemente in linea con le aspettative in termini di costi.

**PB:** Per quanto riguarda la Product Baseline<sub>G</sub>, il valore di EAC<sub>G</sub> si è sempre mantenuto entro i valori accettabili e poco al di sotto del valore ideale, corrispondente al preventivo iniziale (BAC). Questo indica che il progetto<sub>G</sub> è in linea con le aspettative in termini di costi, e che anzi le spese previste sono sempre state leggermente al di sotto della stima iniziale di 12.950,00 €.

Anche relativamente all'EAC dell'ultimo sprint<sub>G</sub> (undicesimo) che corrisponde all'effettiva spesa sostenuta dal gruppo, il valore registrato è di 12.510,00 €, e pertanto il gruppo può affermare di aver risparmiato 440,00 € rispetto al budget iniziale.

#### 4.1.2 MPC05 - Planned Value (PV) & MPC04 - Earned Value (EV)

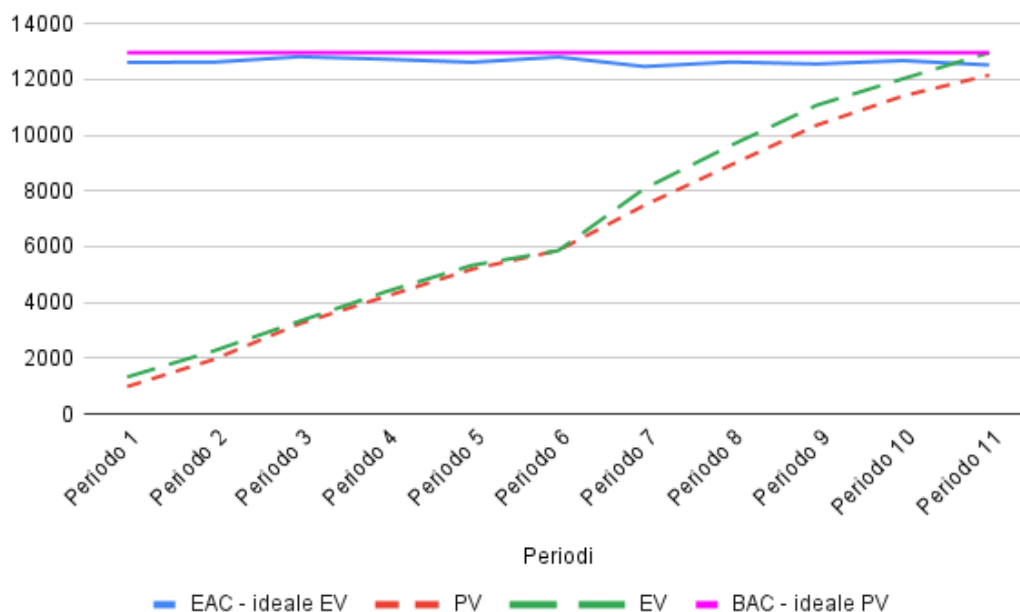


Figure 2: Grafico a linee delle metriche EV e PV

**RTB:** Il grafico mostra la curva del valore guadagnato (Earned Value), e la curva del valore pianificato (Planned Value). Come si può notare osservando la figura le due curve sono molto vicine, questo indica che il lavoro effettivamente svolto è conforme alla pianificazione; e nello specifico quella dell'EV è sempre leggermente superiore a quella del PV<sub>G</sub>. Ciò indica che il progetto<sub>G</sub> sta producendo un valore maggiore rispetto a quanto pianificato, e quindi che il lavoro svolto è leggermente al di sopra delle aspettative.

**PB:** In questo caso le osservazioni già fatte per la RTB<sub>G</sub> si possono applicare anche alla PB<sub>G</sub>: il lavoro effettivamente svolto (EV - Earned Value) è sempre stato conforme a quello pianificato (PV - Planned Value), e nello specifico il valore dell'EV è sempre stato leggermente superiore a quello del PV<sub>G</sub>, confermando che il progetto<sub>G</sub> ha prodotto un valore maggiore rispetto a quanto pianificato.

#### 4.1.3 MPC03 - Actual Cost (AC) & MPC02 - Estimate to Complete (ETC)

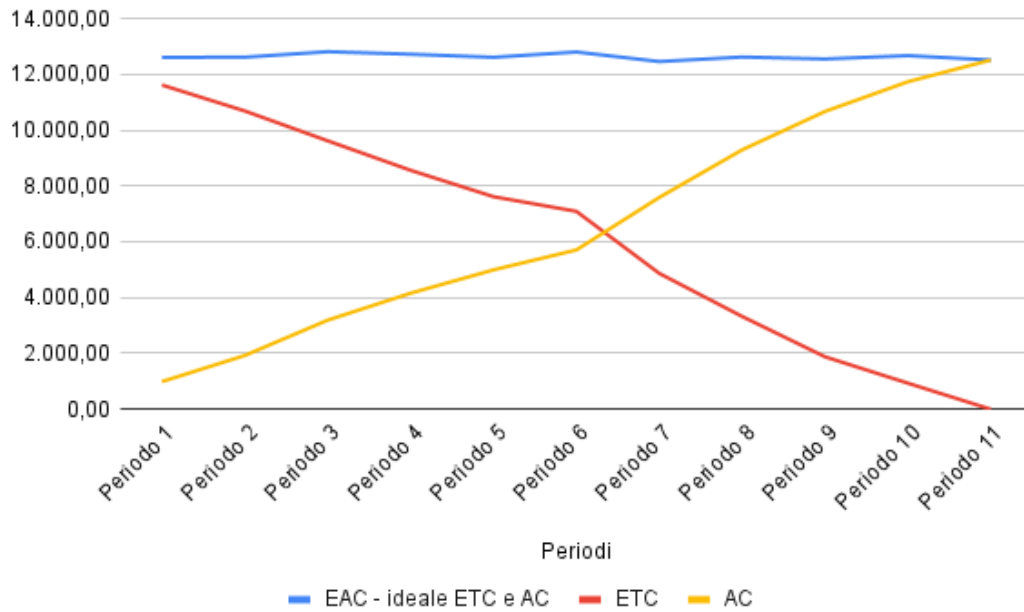


Figure 3: Grafico a linee delle metriche AC e ETC

**RTB:** Il grafico rappresenta l'Actual Cost (AC), ovvero i costi sostenuti per portare il progetto<sub>G</sub> al suo stato corrente, e l'Estimate to Complete (ETC), cioè la stima del costo rimanente da sostenere per completare il progetto<sub>G</sub> per ogni periodo di misurazione (termine sprint<sub>G</sub>). Entrambe le metriche hanno valore ideale inferiore all'EAC, che viene rispettato in ogni iterazione.

Ovviamente l'ETC tende a diminuire al progredire del progetto<sub>G</sub> poiché questo si avvicina alla sua conclusione, mentre l'AC mostra una crescita proporzionale e inversa rispetto all'ETC, in linea con le aspettative economiche del progetto<sub>G</sub>.

**PB:** L'andamento lineare delle due metriche emerso in fase RTB<sub>G</sub> è riconfermato dal grafico anche in fase PB<sub>G</sub>, e testimonia che al crescere delle spese sostenute (AC - Actual Cost), la stima delle spese da sostenere (ETC - Estimate to Complete) è sempre diminuita in maniera proporzionale e inversa.



#### 4.1.4 MPC08 - Cost Performance Index (CPI)

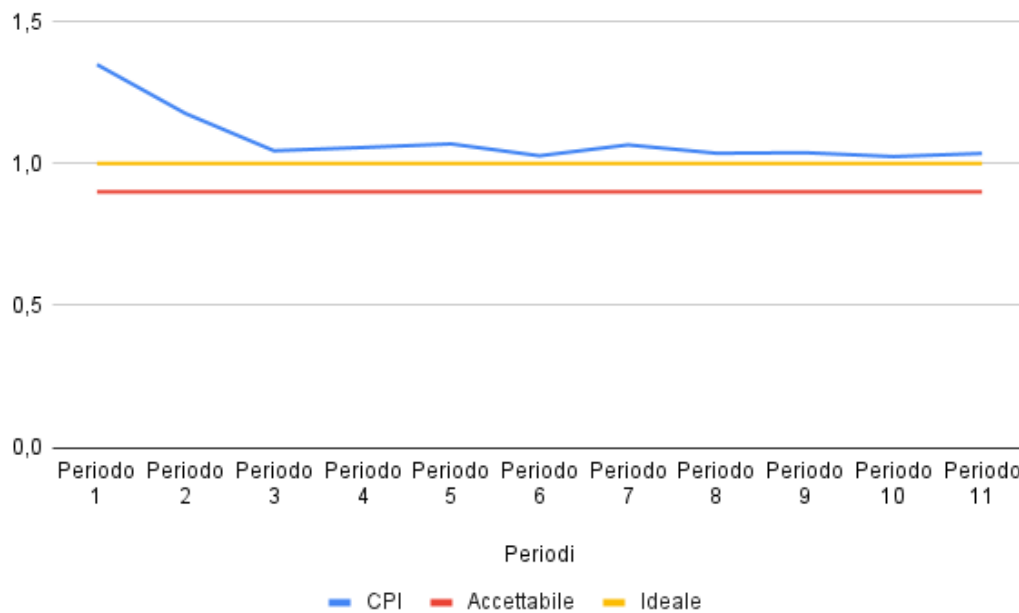


Figure 4: Grafico a linee della metrica CPI

**RTB:** Il Cost Performance Index (CPI) è una metrica che indica quanti obiettivi sono stati raggiunti rispetto alle spese sostenute in un certo momento del progetto<sub>G</sub>. La metrica è data dal rapporto fra il valore guadagnato (EV) e i costi sostenuti (AC), e infatti il suo valore ideale deve essere maggiore o pari a 1. Il valore accettabile invece è stato scelto essere maggiore o uguale a 0.9.

Il grafico mostra che il CPI<sub>G</sub> ha avuto valore strettamente maggiore di 1 per tutto il progetto<sub>G</sub>, assestandosi fra il terzo e il sesto sprint<sub>G</sub> a un valore pari a circa 1.06. Questo indica che il progetto<sub>G</sub> ha prodotto un valore maggiore rispetto ai costi sostenuti.

**PB:** Anche nel corso della seconda fase del progetto<sub>G</sub> la metrica CPI<sub>G</sub> ha continuato il suo andamento, mantenendosi su un valore di poco superiore a 1.00, ovvero al di sopra della soglia del valore ideale. Questo indica che per l'intera durata della seconda fase del progetto<sub>G</sub> il gruppo ha prodotto un valore maggiore rispetto ai costi sostenuti.

#### 4.1.5 MPC07 - Cost Variance (CV) & MPC06 - Schedule Variance (SV)

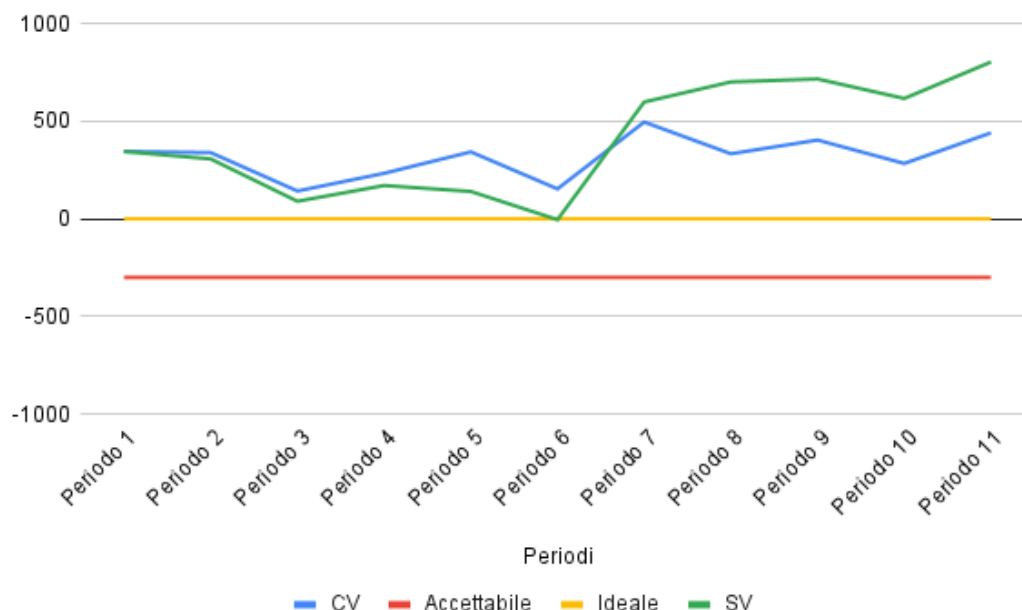


Figure 5: Grafico a linee delle metriche CV e SV

**RTB:** Il grafico mostra l'andamento della Cost Variance (CV) e della Schedule Variance (SV), che rappresentano rispettivamente: la differenza tra il valore guadagnato (EV) e i costi sostenuti (AC) e la differenza tra il valore guadagnato (EV) e il valore pianificato (PV).

La  $CV_G$  resta sempre positiva per l'intera durata del progetto<sub>G</sub> fino al sesto sprint<sub>G</sub>, e ciò indica che le spese effettuate sono inferiori rispetto al valore prodotto dal team nel corso della prima parte del progetto<sub>G</sub>.

La  $SV_G$  mostra un andamento positivo fino al quinto sprint<sub>G</sub>, e ciò indica che il gruppo ha saputo produrre un valore maggiore di quanto pianificato nel corso delle iterazioni effettuate. Invece per il sesto sprint<sub>G</sub> la  $SV_G$  è andata ad un valore non positivo.

**PB:** Nel corso della seconda fase del progetto<sub>G</sub> la metrica  $CV_G$  (Cost Variance) ha, anche in questo caso, mostrato un andamento nettamente positivo, dimostrando che le spese sostenute sono sempre state inferiori rispetto al valore effettivamente prodotto dal team.

Per quanto riguarda invece la metrica  $SV_G$  (Schedule Variance), il suo andamento è stato molto positivo (e in realtà anche significativamente superiore alla  $CV_G$ ) per l'intera durata della seconda fase del progetto<sub>G</sub>. Questo indica che il gruppo ha saputo produrre un valore maggiore di quanto pianificato nel corso delle iterazioni effettuate.

Dall'osservazione delle due metriche messe a confronto emerge però che anche per la seconda fase del progetto<sub>G</sub>, il gruppo ha spesso stimato preventivi di spesa superiori rispetto ai consuntivi effettivamente registrati, fallendo nel rimediare a questa mancanza già osservata nella prima fase del progetto<sub>G</sub>.

## 4.2 Qualità di processo - Sviluppo

### 4.2.1 MPC10 - Requirements Stability Index (RSI)

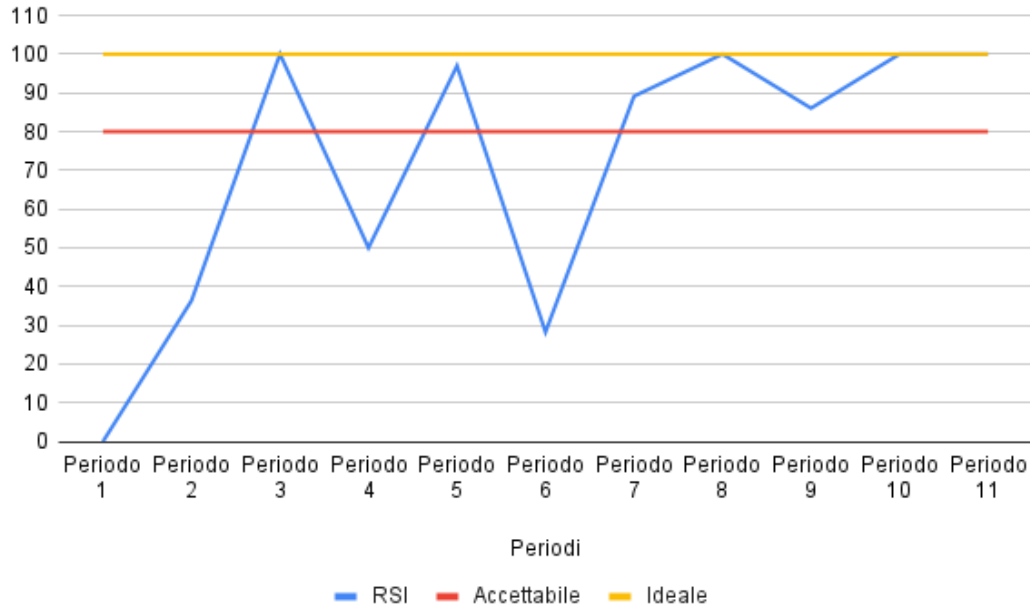


Figure 6: Grafico a linee della metrica RSI

**RTB:** La metrica  $RSI_G$  rappresenta un indice di stabilità dei requisiti individuati nel corso del progetto<sub>G</sub>. Questo indice è in percentuale e assume valore tanto più alto quanto maggiore è la stabilità dei requisiti. In altre parole, il valore è alto se pochi requisiti sono cambiati rispetto all'ultima misurazione, mentre è basso se molti requisiti sono stati modificati, aggiunti o rimossi rispetto all'ultima misurazione.

Come si può evincere dal grafico, l'RSI è partito da valore nullo (in maniera coerente rispetto alla formula usata per il calcolo di tale metrica, poiché nel primo sprint<sub>G</sub> sono stati definiti i primi requisiti); crescendo nella seconda iterazione (mentre il primo tentativo di identificazione dei requisiti era ancora in corso); e ha poi mostrato un andamento altalenante, con picchi di stabilità nel terzo e quinto periodo e cali significativi (inferiori al valore accettabile) in corrispondenza della terza e sesta iterazione. In entrambi questi casi il valore basso è seguito a significative modifiche dei requisiti, successive a degli incontri di chiarimento organizzati con il professor Cardin, che hanno evidenziato degli errori nell'approccio del gruppo all'analisi dei requisiti.

Complessivamente il gruppo si aspetta una maggiore stabilità in futuro, in caso di un buon esito della revisione  $RTB_G$ , mentre potrebbero essere necessari ulteriori significativi cambiamenti in caso di esito negativo.

**PB:** Dall'analisi del grafico della metrica  $RSI_G$  (Requirements Stability Index) emerge che il valore di stabilità dei requisiti è sempre stato entro la soglia accettabile per tutta la durata della seconda fase del progetto<sub>G</sub>, con delle leggere discese (sempre superiori alla soglia di accettazione) in corrispondenza del settimo e nono periodo, in cui si sono concentrati i maggiori sforzi da parte del gruppo di correggere le imprecisioni individuate nell'*Analisi\_dei\_Requisitivi1.0.0* dal professor Cardin.

### 4.3 Qualità di processo - Gestione dei processi

#### 4.3.1 MPC16 - Rischi non previsti



Figure 7: Grafico a linee della metrica "Rischi non previsti"

**RTB:** Nel corso del progetto<sub>G</sub> il team ha senz'altro avuto modo di affrontare svariati dei rischi emersi durante l'analisi dei rischi, tuttavia non sono stati riscontrati rischi non previsti nel corso dei primi sei periodi del progetto<sub>G</sub>.

**PB:** Anche durante la seconda fase del progetto<sub>G</sub> il gruppo non ha riscontrato alcun rischio<sub>G</sub> non previsto.

#### 4.3.2 MPC17 - Efficienza temporale (ET)

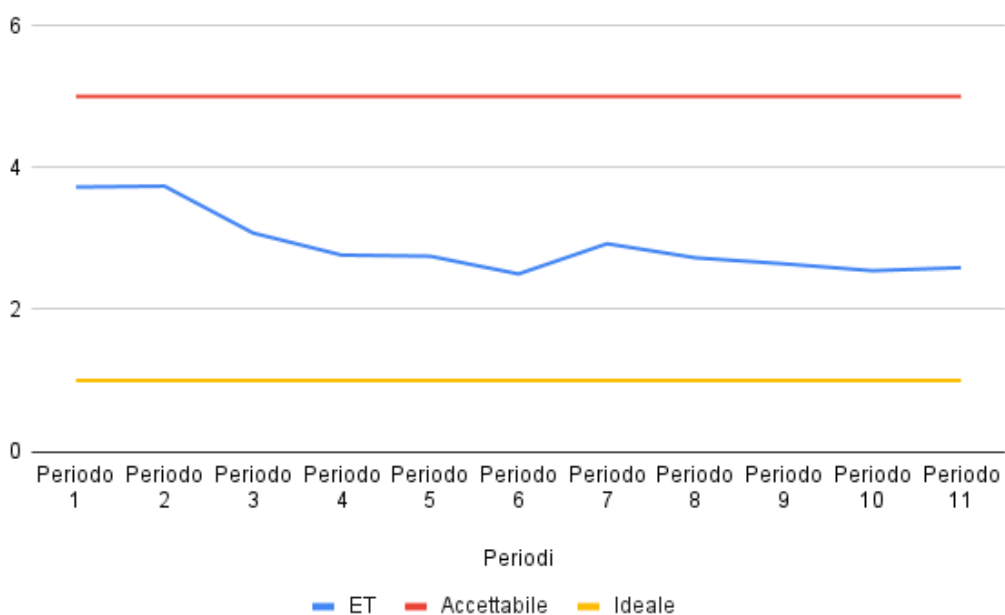


Figure 8: Grafico a linee della metrica ET

**RTB:** La metrica di efficienza<sub>G</sub> temporale è data dal rapporto fra il numero di ore totali e produttive per ogni sprint<sub>G</sub>, e indica quante delle ore dedicate al progetto<sub>G</sub> sono state effettivamente usate per la produzione di materiale (software o di altra natura), rispetto alle ore effettivamente consumate dai membri del gruppo. Le ore produttive sono state accuratamente rendicontate per l'intera durata del progetto<sub>G</sub>, mentre quelle totali (che includono anche le produttive), sono state misurate in maniera più precisa possibile, ma senza pretese di assoluta esattezza, per via della natura di queste ultime.

Dal grafico emerge che la ET si mantiene, per l'intera durata delle prime sei iterazioni del progetto<sub>G</sub>, al di sotto del valore massimo accettabile (pari a 5.0), ma al di sopra del valore ideale (pari a 1.0).

**PB:** Da notare che rispetto alla prima fase del progetto<sub>G</sub>, la formula di calcolo della metrica ET è stata modificata invertendo numeratore e denominatore, per renderla più facilmente interpretabile.

Detto ciò, per la seconda fase del progetto<sub>G</sub> il valore della metrica si è sempre attestato fra 2.5 e 3.0, in linea con le aspettative, ma senza registrare un miglioramento rispetto alle ultime iterazioni della prima fase.

## 4.4 Qualità di processo - Documentazione

### 4.4.1 MPC01 - Errori ortografici

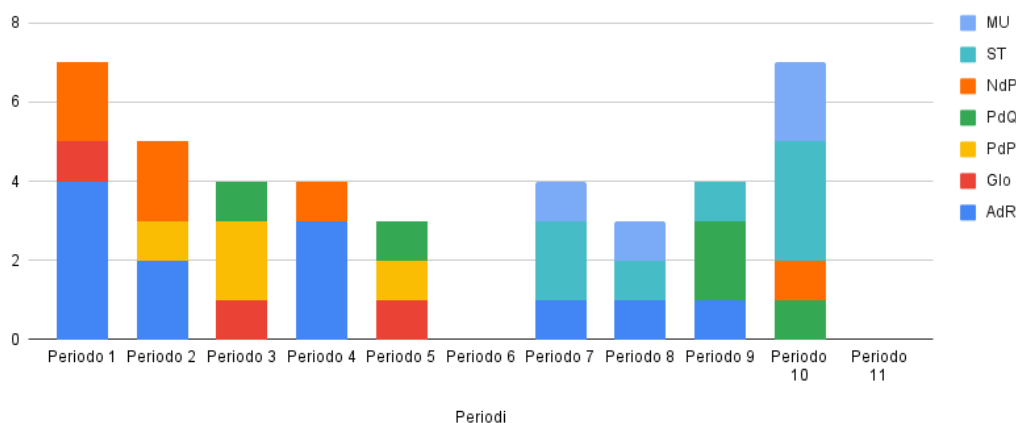


Figure 9: Grafico a linee della metrica "Errori ortografici"

**RTB:** La metrica rappresenta un indice della qualità della documentazione<sub>G</sub> e fa riferimento al numero di errori ortografici individuati in fase di verifica dei documenti prodotti.

Sebbene il valore ideale non sia mai rispettato nel corso dei primi 5 sprint<sub>G</sub> per tutta la documentazione<sub>G</sub>, è bene evidenziare che il team ha mantenuto comunque dei valori bassi, che hanno presentato dei prevedibili aumenti in fasi di grande produzione di documentazione<sub>G</sub>. L'obiettivo del gruppo è comunque stato quello di consegnare al termine del sesto periodo dei documenti privi di errori ortografici, e ciò è stato raggiunto, dato che almeno da parte nostra non ne sono stati individuati.

**PB:** Da notare che, rispetto alla prima fase del progetto<sub>G</sub>, lo stile del grafico è stato modificato da grafico a linee a grafico a barre, per rendere più chiara la lettura dei dati.

Anche nel corso di questa seconda fase del progetto<sub>G</sub> il gruppo ha mantenuto un numero di errori ortografici non conforme al valore accettabile per quasi tutte le iterazioni, riuscendo però a consegnare al termine del progetto<sub>G</sub> dei documenti quanto più possibile privi di errori ortografici. Anche in questo caso sono stati registrati valori più elevati in corrispondenza di periodi di grande produzione di documentazione<sub>G</sub> (soprattutto nel 10 sprint<sub>G</sub>).

#### 4.4.2 MPC11 - Indice Gulpease

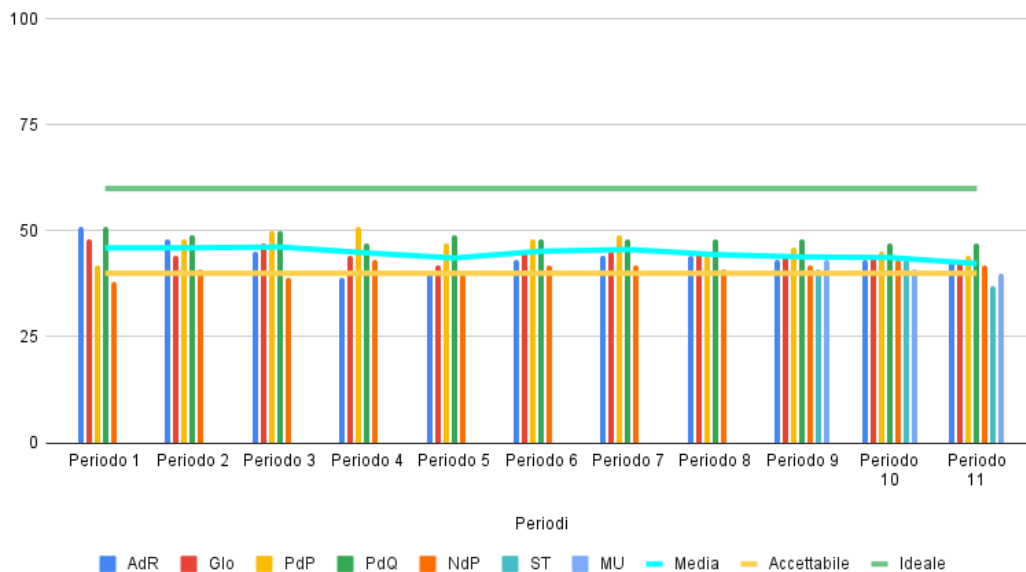


Figure 10: Grafico a linee della metrica "Indice di Gulpease"

**RTB:** Emerge da un'analisi del grafico che il valore di accettazione dell'indice di Gulpease (valore ideale *geq* 60; valore accettabile *geq* 40), è stato quasi sempre rispettato in tutti i documenti redatti per cui si è scelto di misurarlo; con alcune eccezioni per il documento *Analisi dei Requisiti* (nel quarto periodo), e per il documento *Norme di Progetto<sub>G</sub>* (nella prima e nella terza iterazione).

Appare tuttavia chiaro che il gruppo ha avuto non poche difficoltà a raggiungere il valore ideale per questa metrica, poiché è stata sottovalutata la difficoltà dello scrivere documentazione<sub>G</sub> chiara e con linguaggio semplice.

A questo proposito il gruppo si porrà in futuro l'obiettivo di migliorare il valore di questa metrica, cercando di esprimere i concetti con frasi più brevi e semplici.

**PB:** Da notare che lo stile del grafico è stato modificato da grafico a linee a grafico a barre e linee, per rendere più chiara l'interpretazione dei dati.

Relativamente alla seconda fase del progetto<sub>G</sub> il gruppo ha registrato un andamento della metrica analogo a quello della prima fase, con un valore leggermente più basso rispetto alla prima fase.

Malgrado ciò il valore accettabile risulta rispettato sia per la media, che per quasi tutti i singoli documenti, ad eccezione del documento *Specifiche Tecniche v1.0.0*, per la quale, nell'undicesimo periodo si è registrato un valore pari a 37 (leggermente al di sotto della soglia accettabile).

## 4.5 Qualità di processo - Gestione della qualità

### 4.5.1 MPC15 - Metriche di qualità soddisfatte

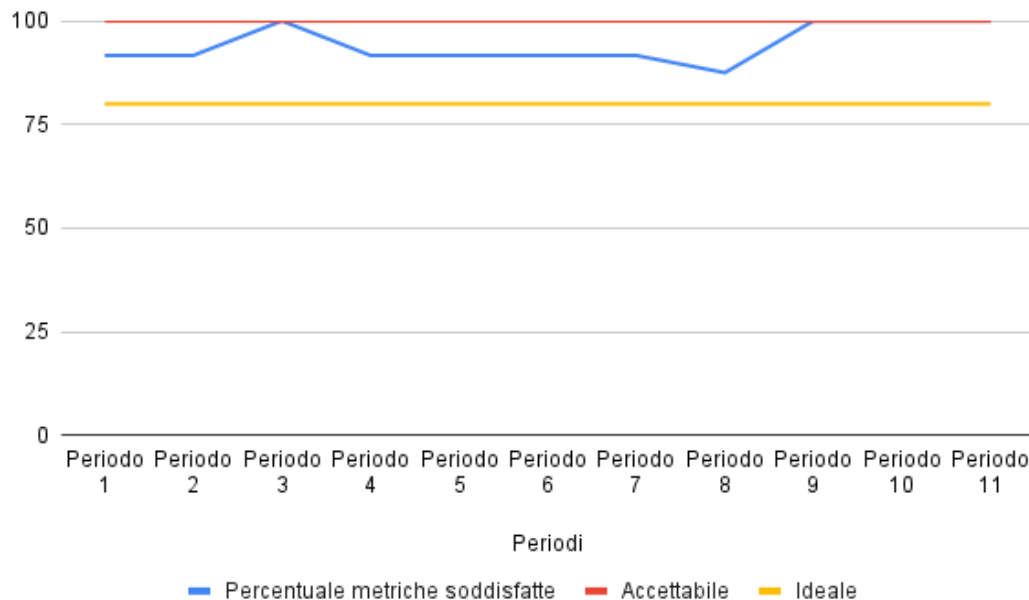


Figure 11: Grafico a linee della metrica "metriche di qualità soddisfatte"

**RTB:** Questa metrica rispecchia la capacità del team di rispettare i valori accettabili delle altre metriche misurata nel corso del progetto<sub>G</sub>. Dall'analisi del suo grafico emerge che il gruppo è sempre riuscito a garantire il rispetto di più dell'90% delle metriche misurate. Ciò è un segnale chiaramente positivo che dimostra un buon lavoro del gruppo dal punto di vista della gestione della qualità.

Detto ciò è però bene notare che il gruppo, in solo uno dei precedenti sei periodi, ha raggiunto il valore ideale (100%), quindi ci sono comunque importanti margini di miglioramento da perseguire.

Un'ultima nota va aggiunta per specificare che il gruppo ha scelto di considerare la metrica relativa all'indice di Gulpease, che si applica a cinque documenti distinti, come una metrica unica facendo la media dei cinque valori misurati.

**PB:** Anche nel corso della seconda fase del progetto<sub>G</sub> il gruppo ha sempre rispettato il valore accettabile della metrica, anche se il minimo storico (pari a 87,5%) è stato registrato in corrispondenza della ottava iterazione.

Anche in questo caso il gruppo non è riuscito a mantenere un valore costante pari al 100%, ma ha comunque raggiunto tale valore ideale in corrispondenza degli ultimi due sprint<sub>G</sub> del progetto<sub>G</sub>.

## 4.6 Qualità di processo - Analisi dei requisiti

### 4.6.1 MPC09 - Requisiti Obbligatori Soddisfatti (ROS)

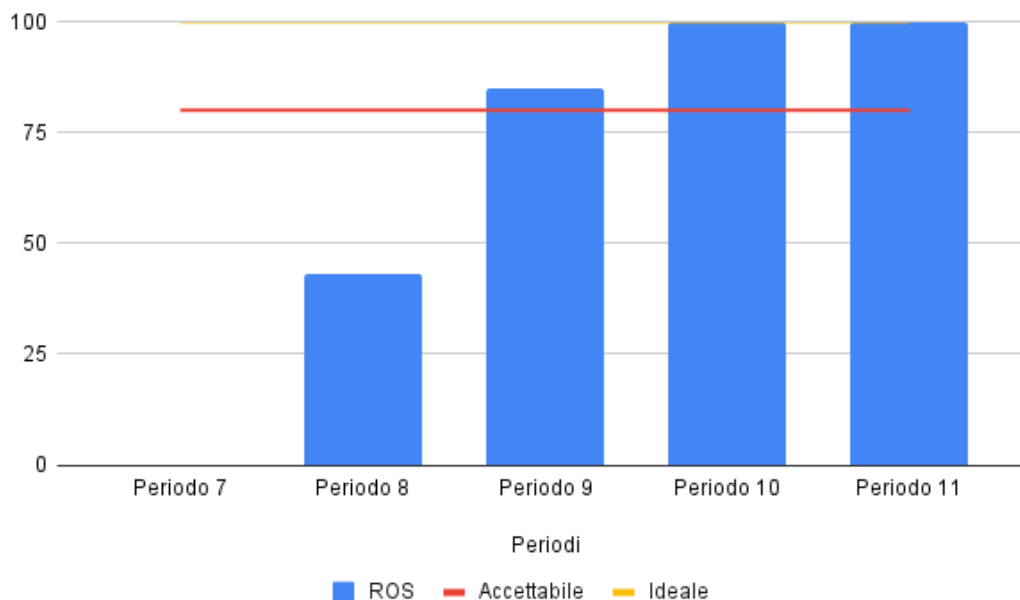


Figure 12: Grafico a linee della metrica ROS

**PB:** Questa metrica riflette la percentuale di requisiti obbligatori soddisfatti durante ogni iterazione. Analizzando il grafico emerge che nel periodo 7 (in cui il gruppo aveva iniziato la progettazione ma non la codifica) il valore della metrica non è stato considerato. Dal periodo successivo sono iniziate le misurazioni, che hanno mostrato un valore inizialmente basso (circa 40%) in corrispondenza della ottava iterazione (poiché il codice era ancora incompleto), e un andamento crescente, con valori positivi nelle successive tre iterazioni.

La costante progressione verso il 100% dei requisiti obbligatori soddisfatti, culminata nel raggiungimento del valore ideale, è un segnale estremamente positivo. Dimostra una solida comprensione delle esigenze del progetto<sub>G</sub> e un'efficace esecuzione da parte del team nel tradurre tali requisiti in funzionalità complete.



## 4.7 Qualità di processo - Verifica

### 4.7.1 MPD04 - Line Coverage

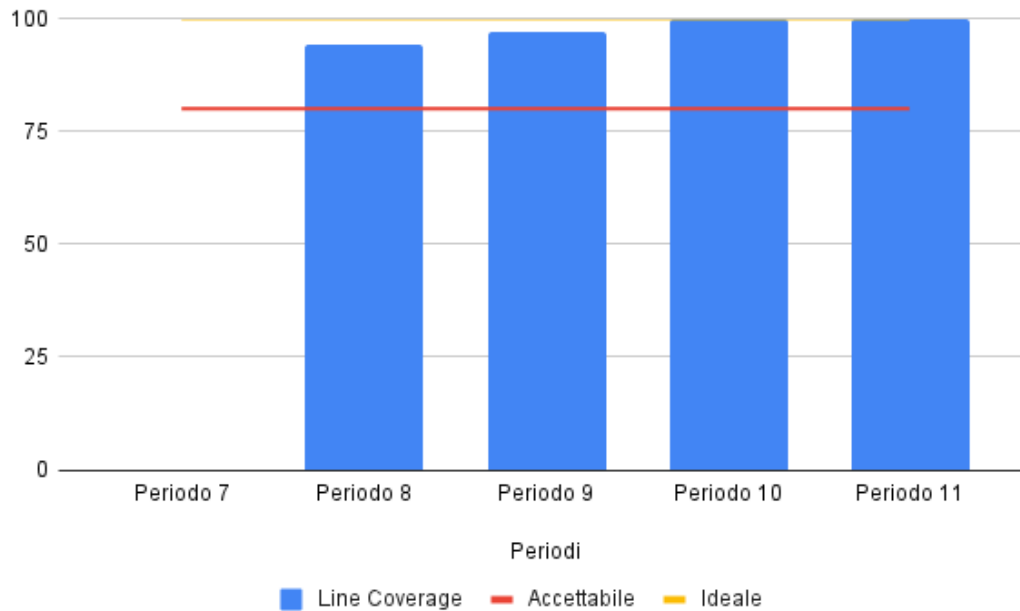


Figure 13: Grafico a linee della metrica "Line coverage"

**PB:** La metrica rappresenta la percentuale di linee di codice coperte dai test<sub>G</sub> automatici. Il valore ideale è 100%, mentre il valore accettabile è 90%.

Il grafico mostra che il valore di questa metrica è sempre stato al di sopra del valore accettabile, e ha raggiunto il valore ideale in prossimità della revisione PB<sub>G</sub>.

Da notare che non è presente alcun valore registrato in corrispondenza della settima iterazione poiché nessun test<sub>G</sub> era ancora stato implementato durante quest'ultima.

#### 4.7.2 MPD05 - Branch Coverage

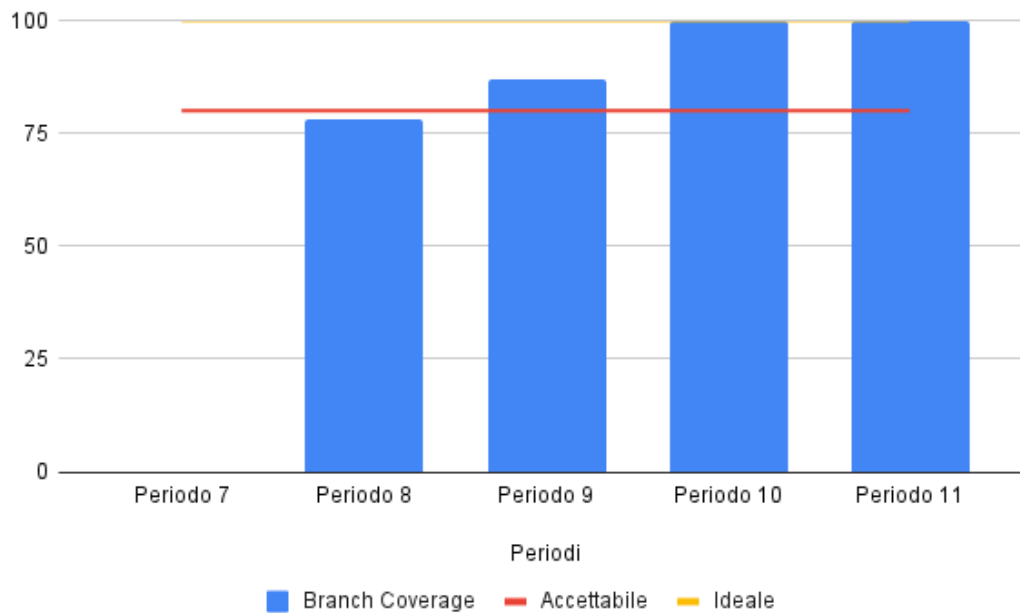


Figure 14: Grafico a linee della metrica "Branch coverage"

**PB:** La metrica rappresenta la percentuale di rami di codice coperti dai test<sub>G</sub> automatici. Il valore ideale è 100%, mentre il valore accettabile è 90%.

Il grafico mostra che il valore di questa metrica è sempre stato al di sopra del valore accettabile, e ha raggiunto il valore ideale in prossimità della revisione PB<sub>G</sub>.

Da notare che non è presente alcun valore registrato in corrispondenza della settima iterazione poiché nessun test<sub>G</sub> era ancora stato implementato durante quest'ultima.

### 4.7.3 MPD09 - Linee di codice per metodo

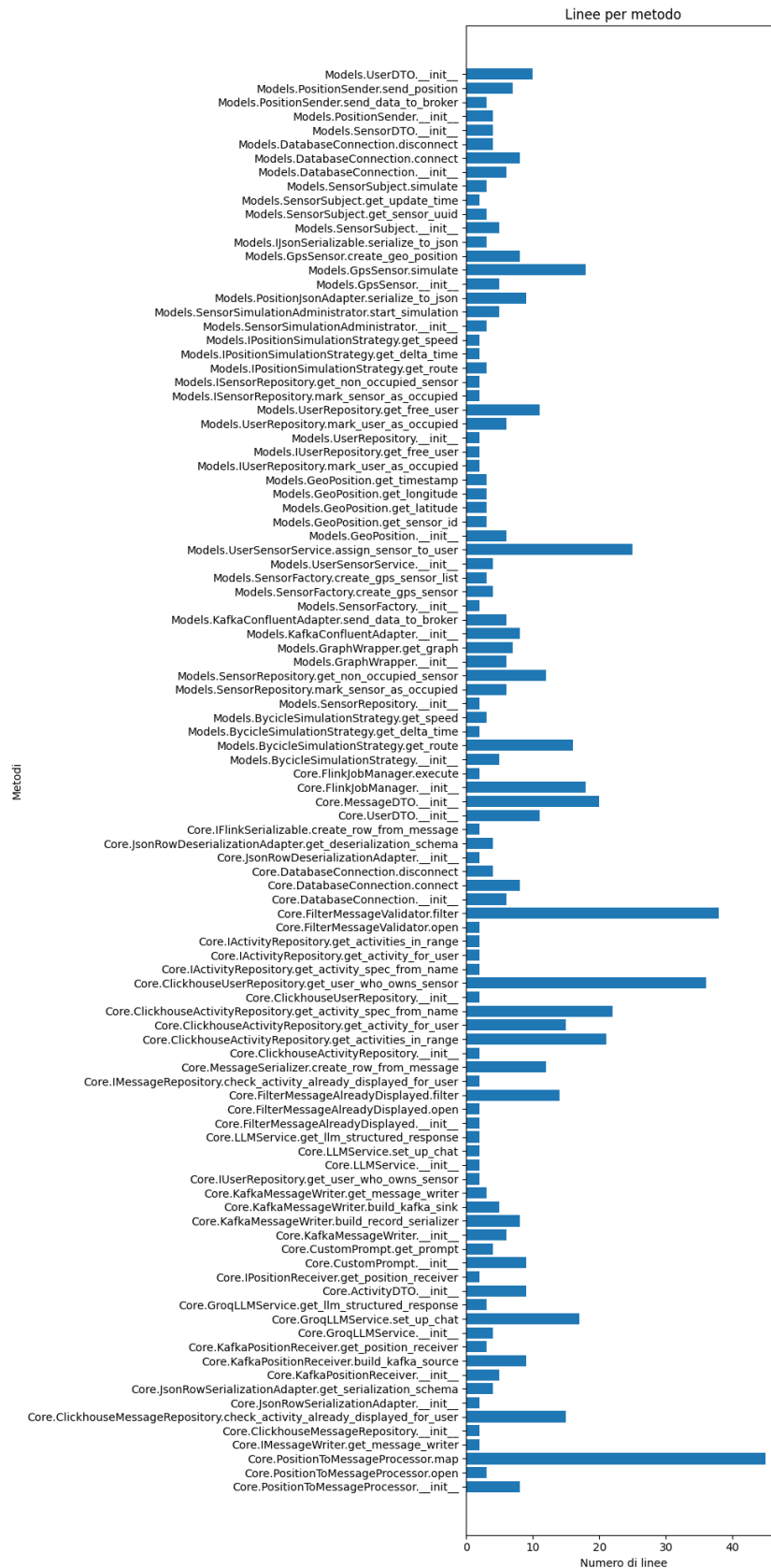


Figure 15: Grafico a barre della metrica "Linee di codice per metodo"

**PB:** Questo grafico rappresenta il numero di linee di codice per ciascun metodo scritto durante la fase di codifica.

Rispetto alla maggior parte delle altre metriche il grafico non rappresenta un andamento temporale, ma mostra il numero di linee di codice per ogni metodo allo stato corrente del prodotto software; ciò perché non sarebbe stato possibile rappresentare in maniera immediata e facilmente leggibile tale metrica altrimenti.

In ogni caso dal grafico si evince che il valore ideale (inferiore a 25) è quasi sempre stato rispettato, ad esclusione di tre casi, in cui è comunque stato registrato un valore accettabile (inferiore a 50).

#### 4.7.4 MPD10 - Attributi per classe

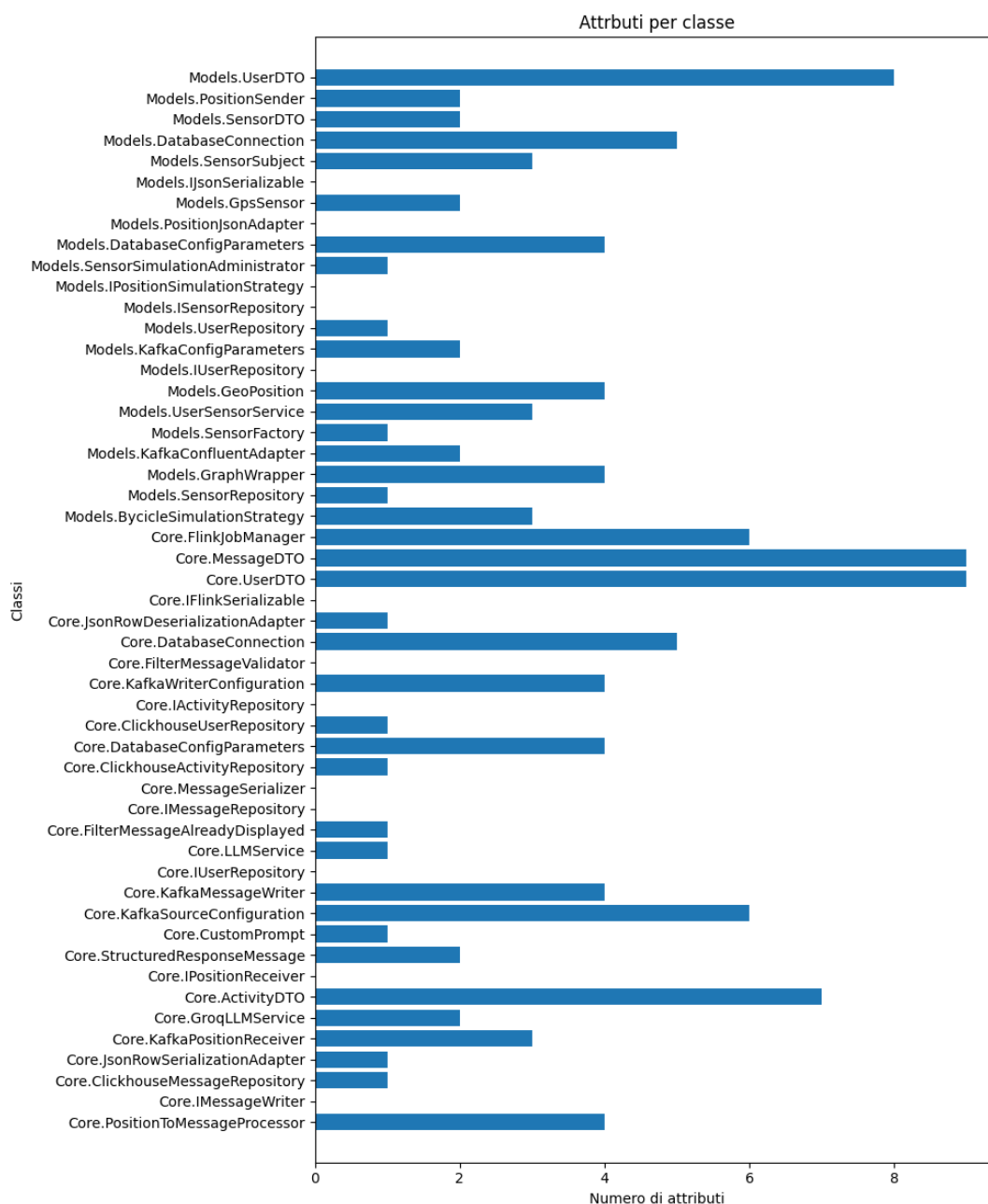


Figure 16: Grafico a barre della metrica "Attributi per classe"

**PB:** Questo grafico rappresenta il numero di attributi per ogni classe scritta durante la fase di codifica. Rispetto alla maggior parte delle altre metriche il grafico non rappresenta un andamento temporale, ma

mostra il numero di attributi per ogni classe allo stato corrente del prodotto software; ciò perché non sarebbe stato possibile rappresentare in maniera immediata e facilmente leggibile tale metrica altrimenti. Analizzando il grafico si evince che il valore ideale (inferiore a 5) è sempre stato rispettato, ad esclusione di un caso, in cui è comunque stato registrato un valore accettabile (inferiore a 7).

#### 4.7.5 MPD11 - Structure Fan IN

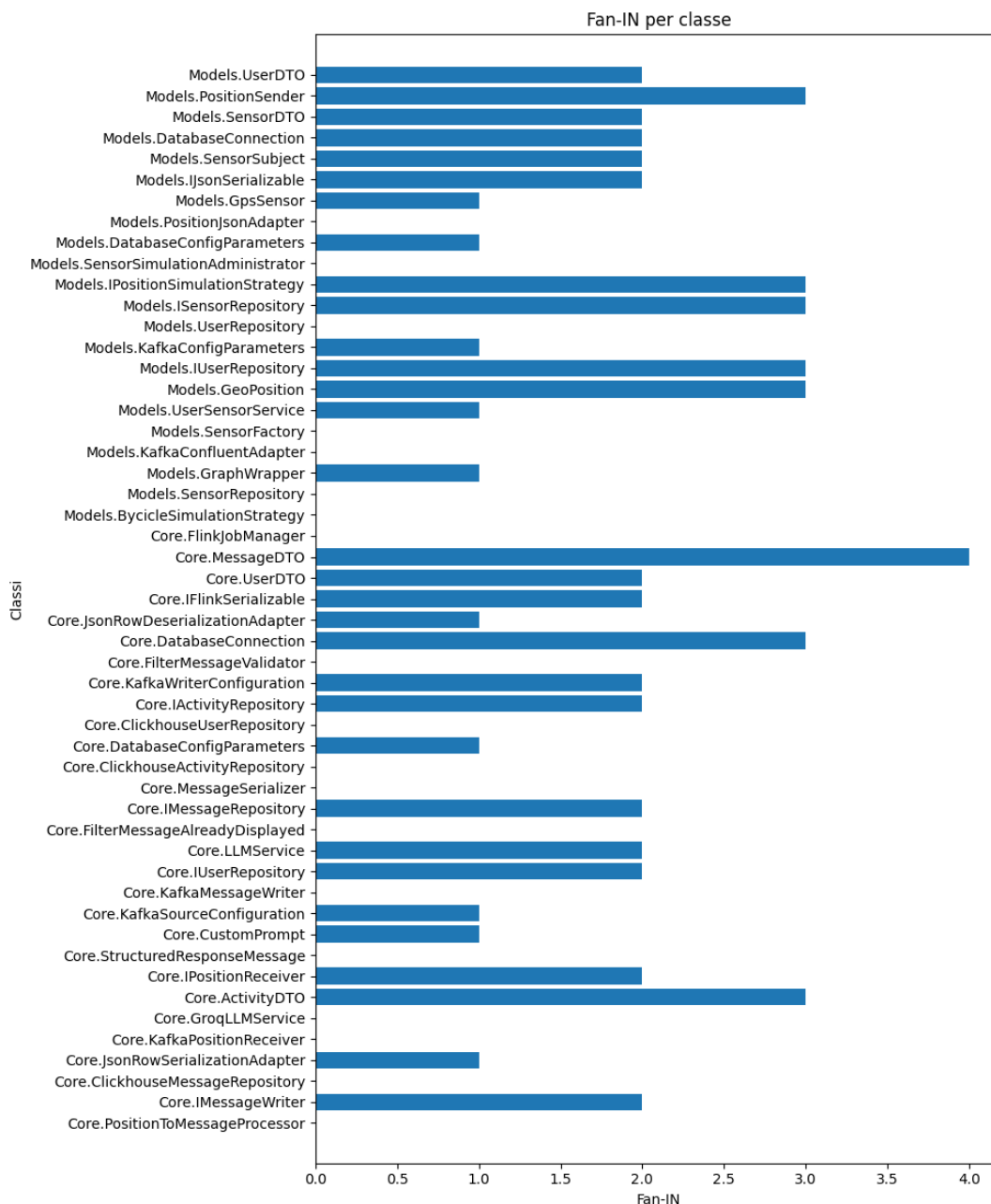


Figure 17: Grafico a barre della metrica "Structure Fan IN"

**PB:** Questo grafico rappresenta per ogni classe il numero di altre classi che dipendono da essa. Rispetto alla maggior parte delle altre metriche il grafico non rappresenta un andamento temporale, ma mostra il valore di fan-IN per ogni classe allo stato corrente del prodotto software; ciò perché non sarebbe stato possibile rappresentare in maniera immediata e facilmente leggibile tale metrica altrimenti. Dall'analisi del grafico emerge che la fan-IN è sempre stata inferiore a 5 e per molti moduli si è registrato un valore nullo. Questa metrica non ha un valore ideale, poiché un alta fan-IN indica che una classe è

molto utilizzata da altre classi, e quindi è un buon segnale di riutilizzo del codice; ma allo stesso tempo un valore elevato indica che una classe crea molto accoppiamento essendo molto utilizzata, e quindi è un segnale negativo.

In generale i valori riscontrati non sono troppo elevati, e sono in linea con le aspettative del gruppo.

#### 4.7.6 MPD12 - Structure Fan OUT

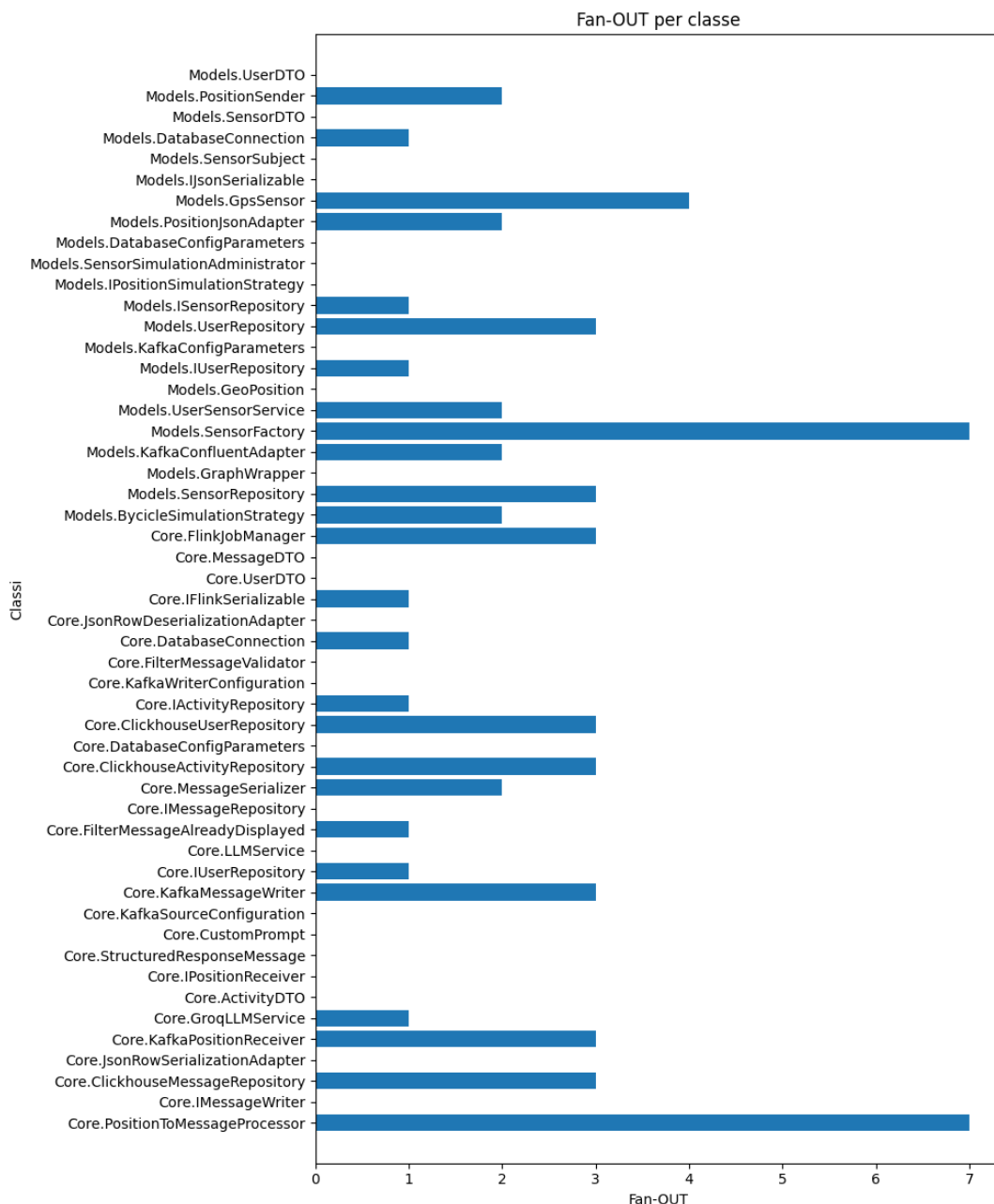


Figure 18: Grafico a barre della metrica "Structure Fan OUT"

**PB:** Questo grafico rappresenta per ogni classe il numero di altre classi la prima chiama o usa. Rispetto alla maggior parte delle altre metriche il grafico non rappresenta un andamento temporale, ma mostra il valore di fan-OUT per ogni classe allo stato corrente del prodotto software; ciò perché non sarebbe stato possibile rappresentare in maniera immediata e facilmente leggibile tale metrica altrimenti. Dall'analisi del grafico emerge che la fan-OUT è sempre stata inferiore a 7 e per molti moduli si è registrato un valore nullo. Questa metrica non ha un valore ideale, poiché un alta fan-OUT indica che una

classe è molto dipendente da altre, cosa che generalmente è negativa (a meno che la classe non sia un controller o orchestratore); ma allo stesso tempo un valore basso, anche se  $p$  indice di buona coesione e indipendenza, può anche indicare che una classe è troppo semplice o superflua.

In generale il valore di fan-OUT ad eccezione di alcuni casi, è stato relativamente basso, in linea con le aspettative del gruppo.

#### 4.7.7 MPC14 - Passed test cases percentage

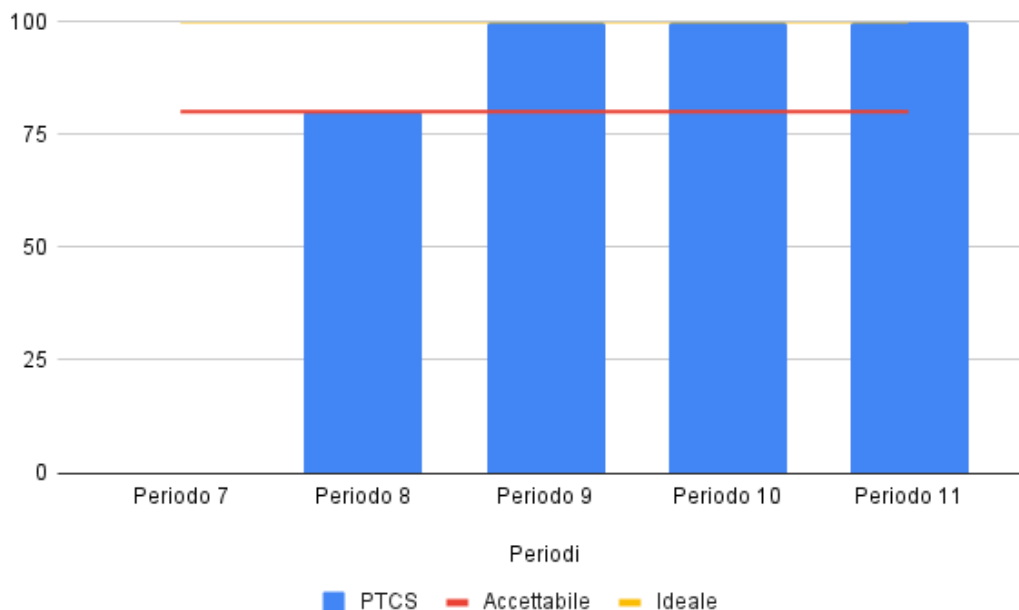


Figure 19: Grafico a linee della metrica "Passed test cases percentage"

**PB:** Questa metrica rappresenta la percentuale di  $test_G$  superati rispetto al totale dei  $test_G$  eseguiti. Il valore ideale è 100%, mentre il valore accettabile è 90%.

Il grafico mostra che il valore di questa metrica è sempre stato al di sopra del valore accettabile, e ha raggiunto il valore ideale in prossimità della revisione PB<sub>G</sub>.

Da notare che non è presente alcun valore registrato in corrispondenza della settima iterazione poiché nessun  $test_G$  era ancora stato implementato durante quest'ultima.

## 5 Metriche non incluse nel cruscotto

Le seguenti metriche, pur essendo parte integrante della valutazione della qualità del prodotto, non sono state incluse nel cruscotto<sub>G</sub> di valutazione delle qualità. La loro natura non si presta infatti a una rappresentazione grafica. Di seguito sono riportati i dettagli relativi a ciascuna metrica, con le rispettive soglie di accettazione e i valori attualmente rilevati.

Metrica	Descrizione	Valore accettazione	Valore attuale
MPD06	Tempo medio di risposta	$\leq 10$ secondi	2 secondi
MPD07	Facilità di utilizzo	$\leq 7$ click	5 click
MPD08	Tempo medio di apprendimento	$\leq 5$ minuti	3
MPD13	Versioni browser supportati	$\geq 80\%$	100%

Table 18: Metriche di qualità non incluse nel cruscotto