

Piano di Qualifica

RAMtastic6

29 aprile 2024



email: ramtastic6@gmail.com

Informazioni sul documento

Versione: 1.0.0
Redattori: Zaupa R. Zambon M. Brotto D.
Verificatori: Brotto D. Visentin S.
Destinatari: T. Vardanega, R. Cardin, Imola Informatica
Uso: Esterno

Registro dei Cambiamenti - Changelog

Versione	Data	Autore	Verificatore	Dettaglio
v 1.0.0	2024-04-29	Zambon M.	Zambon M.	Approvazione e validazione del documento
v 0.3.0	2024-04-28	Zambon M.	Visentin S.	Aggiunti grafici nella sezione 5
v 0.2.1	2024-04-28	Zambon M.	Visentin S.	Stesura sezione 5. (Analisi dei grafici)
v 0.2.0	2024-04-24	Brotto D.	Visentin S.	Aggiornata la sezione dei $Test_G$ di $Sistema_G$ (messa in linea con i ROF corretti nell'Analisi dei Requisiti). Inizio della sezione "Modifiche Migliorative".
v 0.1.3	2024-04-04	Zambon M.	Visentin S.	Cambiate alcuni termini che sono presenti nel glossario per risolvere alcuni problemi con l'automazione dei riferimenti
v 0.1.2	2024-03-26	Zambon M.	Visentin S.	Completata stesura della sottosezione " $Test_G$ di $sistema_G$ " nella sezione testing
v 0.1.1	2024-03-25	Zambon M.	Visentin S.	Iniziata stesura della sottosezione " $Test_G$ di $sistema_G$ " nella sezione testing. Definiti 25/85 test di $sistema_G$
v 0.1.0	2024-03-21	Zambon M.	Brotto D.	Cambiata metrica nella sezione "obiettivi di qualità": "sviluppo" (da "Complessità ciclomatica" a "Linee di codice")
v 0.0.4	2024-03-20	Zambon M.	Brotto D.	Stesura delle prime versioni delle sottosezioni di "Qualità di prodotto"
v 0.0.3	2024-03-13	Zaupa R.	Brotto D.	Stesura delle prime versioni della sezione "Testing" e della sezione "Checklist"
v 0.0.2	2024-03-12	Zaupa R.	Brotto D.	Stesura delle prime versioni delle sottosezioni di "Obiettivi di qualità": "Documentazione" e "Sviluppo"
v 0.0.1	2024-03-11	Zaupa R.	Brotto D.	Prima versione, identificazione delle sezioni principali, stesura della sezione 1 "Introduzione", della sezione introduttiva di "Qualità di $processo_G$ " e di "Qualità di prodotto" e della sottosezione " $Fornitura_G$ "

Indice

1	Introduzione	4
1.1	Scopo del documento	4
1.2	Riferimenti	4
1.2.1	Riferimenti normativi	4
1.2.2	Riferimenti informativi	4
1.3	Codifica delle metriche	5
2	Obiettivi di qualità	6
2.1	Qualità di processo	6
2.1.1	Processi primari	6
2.1.1.1	Fornitura	6
2.1.1.2	Sviluppo	7
2.1.2	Processi di supporto	8
2.1.2.1	Documentazione	8
2.1.3	Processi organizzativi	9
2.2	Qualità di prodotto	9
2.2.1	Adeguatezza funzionale	10
2.2.2	Efficienza	11
2.2.3	Usabilità	11
2.2.4	Affidabilità	12
2.2.5	Manutenibilità	12
2.2.6	Portabilità	13
3	Testing	14
3.1	Test di sistema	14
4	Checklist	20
4.1	Documentazione	20
4.1.1	Struttura	20
4.1.2	Errori ortografici	20
4.1.3	Analisi Dei Requisiti	20
5	Cruscotto di valutazione della qualità	21
5.1	MPC01-EAC (Estimated at Completion)	21
5.2	MPC02-PV (Planned Value) e MPC04-EV (Earned Value)	22
5.3	MPC05-ETC (Estimated to Complete)	22
5.4	MPC06-CV (Cost Variance), MPC07-SV (Schedule Variance) e MPC08-BV (Budget Variance)	23
5.5	MPC15-NCR (Non Calculated Risk)	24
6	Valutazioni per il miglioramento	25
6.1	Valutazione sull'organizzazione	25
6.2	Valutazione sugli strumenti utilizzati	26
6.3	Valutazione sui ruoli	26

1 Introduzione

1.1 Scopo del documento

Il presente documento si propone di definire le metriche e le metodologie di controllo e misurazione necessarie per garantire la qualità del prodotto e del *processo_G*. In particolare, le metriche di valutazione del prodotto sono correlate ai requisiti e alle aspettative del fornitore. Il *Piano di Qualifica_G* è concepito per essere dinamico ed incrementale, in particolar modo per quanto riguarda le metriche descritte e mira a fornire una valutazione il più obiettiva possibile di ciò che è stato realizzato. Le procedure del *Way Of Working_G* devono essere costantemente osservate e migliorate, al fine di garantire che il prodotto soddisfi le aspettative del cliente e mantenga gli standard di qualità richiesti. Eventuali termini tecnici sono definiti all'interno del documento "Glossario Tecnico".

1.2 Riferimenti

1.2.1 Riferimenti normativi

1. Norme di Progetto
2. Presentazione del *capitolato_G* d'appalto C3 - Progetto *Easy Meal_G*:
<https://www.math.unipd.it/~tullio/IS-1/2023/Progetto/C3.pdf>
3. Regolamento del progetto didattico:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/PD2.pdf>

1.2.2 Riferimenti informativi

1. Lezione "*Progettazione software_G (T6)*" del corso di *Ingegneria del software_{GG}*:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T6.pdf>
2. Lezione "*Qualità del software_G (T7)*" del corso di *Ingegneria del software_{GG}*:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T7.pdf>
3. Lezione "*Qualità di processo_G (T8)*" del corso di *Ingegneria del software_G*:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T8.pdf>
4. Lezione "*Verifica e validazione: introduzione (T9)*" del corso di *Ingegneria del software_G*:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T9.pdf>
5. Lezione "*Verifica e validazione: analisi statica (T10)*" del corso di *Ingegneria del software_G*:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T10.pdf>
6. Lezione "*Verifica e validazione: analisi dinamica (T11)*" del corso di *Ingegneria del software_G*:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T11.pdf>
7. Documento "*Dichiarazione impegni v1.2*":
https://github.com/RAMtastic6/Project14/blob/main/documenti/CANDIDATURA/documento_impegni_v1.2.pdf
8. Metriche di progetto (*Earned Value Analysis*):
https://it.wikipedia.org/wiki/Metriche_di_progetto

1.3 Codifica delle metriche

In questa sottosezione verranno definite le metriche che utilizzeremo, utilizzando un codice standardizzato.

Una metrica è identificata dal seguente formato di codice:

$$M[\text{Tipo}][\text{Id}]-[\text{Acronimo}]$$

Dove:

- **M** sta per "Metrica"
- **Tipo** può essere PC (per un *processo_G*) o PD (per un prodotto)
- **Id** rappresenta un identificativo all'interno di una metrica di un certo tipo
- **Acronimo** indica l'acronimo del nome della metrica utilizzata

Per ciascuna metrica verranno fornite descrizioni, valori accettabili e valori preferibili.

2 Obiettivi di qualità

2.1 Qualità di processo

La verifica della qualità si misura sul come sta andando il *processo_G*, si misura l'efficacia e l'efficienza. Dopo aver definito il *processo_G* in esame nel documento di *Norme di Progetto_G*, si deve dunque misurare l'efficacia e l'efficienza, imparando cosa trattenere e cosa correggere. Per farlo sceglieremo delle buone metriche di misurazione di qualità e anche buoni strumenti (serve oggettività e automazione).

2.1.1 Processi primari

2.1.1.1 Fornitura

Per definire le metriche adottate per i *processi primari_G* di *fornitura_G*, andremo a fare riferimento alla tecnica denominata *EVA (Earned Value Analysis)*, consente di calcolare in termini di tempo, denaro speso e valore del lavoro realizzato e di valutare la performance del progetto.

Tra queste individuiamo:

- ***Budget at Completion (BAC)***:
valore previsto iniziale per la realizzazione del progetto
- ***Estimated at Completion (EAC)***:
revisione del valore stimato per la realizzazione del progetto
- ***Planned Value (PV)***:
rappresenta il costo pianificato per realizzare le attività di progetto per la data stabilita (misurato in tempo o in denaro)
- ***Actual cost (AC)***:
rappresenta il costo effettivamente sostenuto alla data stabilita (misurato in tempo o in denaro)
- ***Earned Value (EV)***:
rappresenta il valore delle attività sostenute alla data stabilita (misurato in tempo o in denaro)
- ***Estimated to Complete (ETC)***:
valore stimato per completare le attività rimanenti necessarie per concludere il progetto
- ***Cost Variance (CV)***:
misura la variazione del valore ottenuto (EV) rispetto al costo effettivo (AC)
- ***Schedule Variance (SV)***:
misura la variazione del valore ottenuto (EV) rispetto al costo pianificato (PV)
- ***Budget Variance (BV)***:
misura la variazione dal costo attuale (AC) rispetto al costo atteso (CV)

Come riportato dal documento "*Dichiarazione impegni v1.2*" citato nella sezione 1.2.2 "*Riferimenti_G informativi*", il BAC corrisponde ad un valore di €11.520.

Di seguito vengono riportati gli obiettivi di qualità per tali metriche:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPC01-EAC	Estimated at Completion	$\pm 5\%$ BAC	$= BAC$
MPC02-PV	Planned Value	≥ 0	$\leq BAC$
MPC03-AC	Actual Cost	≥ 0	$\leq EAC$
MPC04-EV	Earned Value	≥ 0	$\leq EAC$
MPC05-ETC	Estimated to Complete	≥ 0	$\leq EAC$
MPC06-CV	Cost Variance	$\geq -7.5\%$	≥ 0
MPC07-SV	Schedule Variance	$\geq -7.5\%$	≥ 0
MPC08-BV	Budget Variance	$\pm 10\%$	$= 0$

Tabella 1: Metriche per i processi di fornitura

2.1.1.2 Sviluppo

Analisi Dei Requisiti Nel caso dell'attività di *Analisi Dei Requisiti_G* andremo a definire le seguenti metriche:

- **Requirements stability index (RSI):**

misura la stabilità dei requisiti nel corso del tempo durante lo sviluppo del progetto. Si misura nel seguente modo:

$$RSI = 1 - \left(\frac{N_{\text{modifiche}}}{N_{\text{requisiti totali}}} \right) \times 100$$

Dove:

- $N_{\text{modifiche}}$ è il numero totale di modifiche apportate ai requisiti durante un periodo di tempo specifico.
- $N_{\text{requisiti totali}}$ è il numero totale di requisiti nel progetto.

Progettazione Nel caso delle attività di progettazione andremo a definire le seguenti metriche:

- **Structural fan-in (SFIN):**

misura la quantità dei moduli che utilizzano un modulo specifico. Un valore alto indica che molte parti del *sistema_G* dipendono da un modulo specifico.

- **Structural fan-out (SFOUT):**

misura il numero di moduli utilizzati da un modulo specifico. Un valore alto indica che il modulo preso in considerazione dipende da molti altri moduli.

Codifica Nel caso dell'attività di *codifica_G* andremo a definire le seguenti metriche:

- **Linee di codice (LOC):**

misura la complessità del *software_G* in base al numero di linee di codice sorgente.

Di seguito vengono riportati gli obiettivi di qualità per tali metriche:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPC09-RSI	Requirements stability index	$\geq 70\%$	100%
MPC10-SFIN	Structural fan-in	-	Massimo
MPC11-SFOUT	Structural fan-out	-	Minimo
MPC12-LOC	Linee di codice	≤ 50.000	≤ 30.000

Tabella 2: Metriche per i processi di sviluppo

2.1.2 Processi di supporto

2.1.2.1 Documentazione

Per il *processo_G* di documentazione andremo a definire le seguenti metriche:

- **Errori ortografici (EO):**
misura la quantità di errori ortografici individuati per documento
- **Indice Gulpease (IG):**
misura la leggibilità di un documento in lingua italiana. Si calcola nel seguente modo:

$$IG = 89 + \frac{300 \times (\text{num. frasi}) - 10 \times (\text{num. lettere})}{\text{num. parole}}$$

dove:

- *num. frasi* è il numero di frasi nel testo,
- *num. lettere* è il numero di lettere nel testo,
- *num. parole* è il numero di parole nel testo.

Di seguito vengono riportati gli obiettivi di qualità per tali metriche:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPC13-EO	Errori ortografici	0	0
MPC14-IG	Indice Gulpease	30-100	50-100

Tabella 3: Metriche per i processi di supporto

2.1.3 Processi organizzativi

Per i *processi organizzativi*_G andremo ad utilizzare le seguenti metriche:

- **Non Calculated Risk (NCR):**
misura la quantità di *rischi*_G non previsti e non stimati
- **Efficienza temporale (ET):**
misura quanto tempo viene impiegato per attività produttive rispetto alle ore individuali:

$$ET = \frac{\text{Ore individuali}}{\text{Ore produttive}}$$

dove:

- *Ore individuali* rappresentano le ore che portano al raggiungimento di obiettivi
- *Ore produttive* rappresentano il tempo totale trascorso come ore di orologio

Di seguito vengono riportati gli obiettivi di qualità per tali metriche:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPC15-NCR	Non Calculated Risk	≤ 4	0
MPC16-ET	Efficienza temporale	≤ 3	≤ 1

Tabella 4: Metriche per i processi organizzativi

2.2 Qualità di prodotto

In questa sezione si discute degli obiettivi che un prodotto *software*_G di qualità dovrebbe avere. Di seguito vengono elencati gli obiettivi di qualità esterni.

- **Adeguatezza funzionale:** si riferisce alla capacità di fornire le funzionalità e le caratteristiche previste permettendo di soddisfare i requisiti funzionali.
- **Efficienza:** si riferisce alla capacità di fornire adeguate prestazioni e a quella di utilizzo delle risorse di *sistema*_G.
- **Usabilità:** si riferisce alla misura all'apprendimento, comprensione e all'operabilità del prodotto da parte dell'utente.
- **Affidabilità:** misura la capacità del prodotto di funzionare correttamente sotto determinate condizioni per un determinato periodo di tempo.
- **Sicurezza:** indica la capacità del *software*_G di proteggere i dati sensibili e prevenire accessi non autorizzati oltre che violazioni della privacy.

Di seguito vengono elencati gli obiettivi di qualità interni.

- **Manutenibilità:** misura la facilità con cui il prodotto può essere modificato, aggiornato e corretto.
- **Portabilità:** misura la facilità con cui il prodotto può essere trasferito da un ambiente all'altro.

2.2.1 Adeguatezza funzionale

Per l'adeguatezza funzionale andremo ad utilizzare le seguenti metriche:

- **Copertura dei requisiti obbligatori (CRO):**
misura la percentuale dei requisiti obbligatori soddisfatti

$$CRO = \frac{N_{ros}}{N_{rot}} \times 100$$

dove:

- N_{ros} rappresentano il numero di requisiti obbligatori soddisfatti
- N_{rot} rappresentano il numero di requisiti obbligatori totali

- **Copertura dei requisiti desiderabili (CRD):**
misura la percentuale dei requisiti desiderabili soddisfatti

$$CRD = \frac{N_{rds}}{N_{rdt}} \times 100$$

dove:

- N_{rds} rappresentano il numero di requisiti desiderabili soddisfatti
- N_{rdt} rappresentano il numero di requisiti desiderabili totali

Di seguito vengono riportati gli obiettivi di qualità per tali metriche:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPD01-CRO	Copertura dei requisiti obbligatori	100%	100%
MPD02-CRD	Copertura dei requisiti desiderabili	$\geq 50\%$	100%

Tabella 5: Metriche per l'adeguatezza funzionale

2.2.2 Efficienza

Per l'efficienza andremo ad utilizzare la seguente metrica:

- **Tempo di risposta media (TM):**
misura il tempo impiegato dal $software_G$ di gestire ed elaborare una richiesta fino al risultato finale.

Di seguito vengono riportati gli obiettivi di qualità per tale metrica:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPD03-TM	Tempo di risposta medio	3 secondi	2 secondi

Tabella 6: Metriche per l'efficienza

2.2.3 Usabilità

Per l'usabilità andremo ad utilizzare le seguenti metriche:

- **Tempo di apprendimento (TA):**
misura il tempo impiegato dall'utente di imparare ad utilizzare le funzionalità del $software_G$.
- **Raggiunta dell'obiettivo (RO):**
misura il numero di iterazioni necessarie all'utente per raggiungere il risultato voluto.
- **Errori dell'utente (EU):**
misura il numero di errori che l'utente compie prima di raggiungere l'obiettivo desiderato.

Di seguito vengono riportati gli obiettivi di qualità per tali metriche:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPD04-TA	Tempo di apprendimento	20 minuti	10 minuti
MPD05-RO	Raggiunta dell'obiettivo	7 click	4 click
MPD06-EU	Errori dell'utente	3	0

Tabella 7: Metriche per l'usabilità

2.2.4 Affidabilità

Per l'affidabilità andremo ad utilizzare la seguente metrica:

- **Failure density (FD):**
misura in percentuale l'affidabilità del $software_G$.

$$FD = \frac{N_{tf}}{N_{te}} \times 100$$

dove:

- N_{tf} rappresentano il numero di $test_G$ falliti
- N_{te} rappresentano il numero di $test_G$ effettuati

Di seguito vengono riportati gli obiettivi di qualità per tale metrica:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPD07-FD	Failure density	10%	0%

Tabella 8: Metriche per l'affidabilità

2.2.5 Manutenibilità

per la manutenibilità andremo ad utilizzare le seguente metrica:

- **Complessità ciclomatica (CC):**
misura la complessità del $software_G$ utilizzando il grafo di controllo del flusso

$$v(G) = e - n + p$$

dove:

- G rappresenta il grafo,
- e rappresenta il numero di archi in G,
- n rappresenta il numero di nodi in G,
- p rappresenta il numero delle componenti connesse da ogni arco.

Di seguito vengono riportati gli obiettivi di qualità per tale metrica:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPD08-CC	Complessità ciclomatica	≤ 15	≤ 10

Tabella 9: Metriche per la manutenibilità

2.2.6 Portabilità

per la portabilità andremo ad utilizzare la seguente metrica:

- **Browser supportati (BS):**

misura in percentuale le versioni dei browser supportati

$$BS = \frac{N_{vbs}}{N_{vbp}} \times 100$$

dove:

- N_{vbs} rappresenta il numero di versioni di browser supportate
- N_{vbp} rappresenta il numero di versioni di browser previste da supportare

Di seguito vengono riportati gli obiettivi di qualità per tale metrica:

Metrica	Nome	Valore Accettabile	Valore Preferibile
MPD09-BS	Browser supportati	100%	100%

Tabella 10: Metriche per la portabilità

3 Testing

In questa sezione vengono esplorate le metodologie di $test_G$ ing e la loro specifica. L'obiettivo è quello di seguire il "Modello a V_G " in cui ad ogni fase di sviluppo corrisponde una tipologia di $test_G$ da eseguire.

I $test_G$ si dividono in:

- **Test di unità:**
vengono eseguiti sulle unità più semplici del codice. Viene fatta corrispondere all'attività di $codifica_G$ (implementation).
- **Test di integrazione:**
vengono eseguiti per verificare la corretta integrazione tra le diverse unità $software_G$. Viene fatta corrispondere all'attività di progettazione.
- **Test di sistema:**
verificano il corretto funzionamento dell'intero $sistema_G$ e, in particolare, che tutti i requisiti individuati siano soddisfatti. Viene fatta corrispondere all'attività di $Analisi Dei Requisiti_G$.
- **Test di accettazione:**
verificano, alla presenza del committente, che il prodotto finale soddisfi tutti i requisiti. Se superati, si può procedere al $rilascio_G$ dello stesso.

3.1 Test di sistema

I $test_G$ di $sistema_G$ sono una fase del $processo_G$ di $test_G$ ing il cui scopo è quello di verificare che il $sistema_G$ $software_G$ rispetti i requisiti specificati nel documento "Analisi Dei Requisiti $_G$ ". Di seguito verranno elencati i vari $test_G$, i quali avranno un codice identificativo, una descrizione, il $requisito_G$ a cui fa riferimento e lo stato del $test_G$.

Codice	Descrizione	Requisito	Stato
TS-01	Verificare che l'utente base possa visualizzare i pasti ordinabili	ROF 1	Non implementato
TS-02	Verificare che il $sistema_G$ possa inviare una notifica se l'utente base sta aggiungendo un piatto che contiene elementi a cui è allergico/intollerante	ROF 2	Non implementato
TS-03	Verificare che l'utente possa visualizzare i pasti con i loro ingredienti e che possa modificarli, aggiungendo o togliendo ingredienti	ROF 3	Non implementato
TS-04	Verificare che l'utente base possa essere in grado di visualizzare il riepilogo di quanto ordinato e che possa confermare oppure cancellare l'ordine	ROF 4	Non implementato
TS-05	Verificare che il $sistema_G$ possa inviare una notifica ($push_G$) di $prenotazione_G$ effettuata ai ristoranti	RDF 5	Non implementato
TS-06	Verificare che l'utente possa inserire le informazioni (data, orario, persone, nomi utenti) per effettuare una $prenotazione_G$	ROF 6	Non implementato

TS-07	Verificare che il <i>sistema_G</i> possa inviare una notifica (<i>push_G</i>) agli amministratori per comunicare la richiesta di <i>prenotazione_G</i> , che può accettare o rifiutare	RDF 7	Non implementato
TS-08	Verificare che l'utente possa essere in grado di cancellare la <i>prenotazione_G</i>	ROF 8	Non implementato
TS-09	Verificare che il <i>sistema_G</i> possa negare la <i>prenotazione_G</i> se il ristorante non possiede abbastanza posti o tavoli	ROF 9	Non implementato
TS-10	Verificare che l'utente base possa visualizzare una lista di ristoranti filtrata per nome, data, luogo o tipologia cucina	ROF 10	Non implementato
TS-11	Verificare che l'utente base possa selezionare un ristorante	ROF 11	Non implementato
TS-12	Verificare che l'amministratore possa visualizzare una lista di prenotazioni in attesa	ROF 12	Non implementato
TS-13	Verificare che l'amministratore possa selezionare una specifica richiesta di <i>prenotazione_G</i> dalla lista visualizzata	ROF 13	Non implementato
TS-14	Verificare che l'amministratore possa accettare la richiesta di <i>prenotazione_G</i> selezionata	ROF 14	Non implementato
TS-15	Verificare che il <i>sistema_G</i> possa aggiungere la <i>prenotazione_G</i> nell'area "prenotazioni" dell'utente	ROF 15	Non implementato
TS-16	Verificare che il <i>sistema_G</i> possa notificare gli utenti coinvolti (nel caso di <i>prenotazione_G</i> collaborativa) dell'accettazione della <i>prenotazione_G</i>	ROF 16	Non implementato
TS-17	Verificare che il <i>sistema_G</i> possa fornire un'interfaccia per consentire all'amministratore di verificare la disponibilità di posti in base alle specifiche della <i>prenotazione_G</i> selezionata	ROF 17	Non implementato
TS-18	Verificare che il <i>sistema_G</i> possa ridurre il numero di posti disponibili in base alle specifiche della <i>prenotazione_G</i> accettata	ROF 18	Non implementato
TS-19	Verificare che l'amministratore possa rifiutare la richiesta di <i>prenotazione_G</i> selezionata	ROF 19	Non implementato
TS-20	Verificare che il <i>sistema_G</i> possa notificare gli utenti coinvolti (nel caso di <i>prenotazione_G</i> collaborativa) del rifiuto della <i>prenotazione_G</i>	ROF 20	Non implementato
TS-21	Verificare che il <i>sistema_G</i> possa creare un canale di comunicazione tra l'utente e l'amministratore del ristorante quando l'utente lo richiede	ROF 21	Non implementato
TS-22	Verificare che l'utente e l'amministratore possano scambiarsi messaggi in modo bidirezionale tramite l'interfaccia di comunicazione	ROF 22	Non implementato
TS-23	Verificare che durante la comunicazione, il <i>sistema_G</i> possa inviare notifiche <i>push_G</i> per informare l'utente e l'amministratore dei nuovi messaggi ricevuti	ROF 23	Non implementato

TS-24	Verificare che la cancellazione della <i>prenotazione_G</i> possa essere effettuata con al massimo un giorno di anticipo rispetto alla data della <i>prenotazione_G</i>	ROF 24	Non implementato
TS-25	Verificare che il <i>sistema_G</i> permetta di pagare il conto in base alla modalità scelta (divisione equa, divisione proporzionale) da chi ha creato la <i>ordinazione_G</i> collaborativa	ROF 25	Non implementato
TS-26	Verificare che l'utente base possa pagare tutto il conto se nessun utente ha pagato	ROF 26	Non implementato
TS-27	Verificare che l'amministratore possa modificare il menù del proprio ristorante, aggiungendo, rimuovendo pietanze e modificando le informazioni del singolo piatto (nome, ingredienti e prezzo)	ROF 27	Non implementato
TS-28	Verificare che la modifica del menù da parte dell'amministratore non causi problemi di sincronizzazione nella visualizzazione, ricerca del menù e nell' <i>ordinazione_G</i> da parte dell'utente base	RDF 28	Non implementato
TS-29	Verificare che l'utente base possa inserire un coupon prima di pagare il conto, che, se applicato, deve far ricalcolare al <i>sistema_G</i> il prezzo del conto	RDF 29	Non implementato
TS-30	Verificare che l'amministratore possa consultare le prenotazioni associate al proprio ristorante	ROF 30	Non implementato
TS-31	Verificare che l'amministratore possa visualizzare i dettagli completi di una specifica <i>prenotazione_G</i>	ROF 31	Non implementato
TS-32	Verificare che l'amministratore possa visualizzare lo stato delle ordinazioni associate alla <i>prenotazione_G</i>	ROF 32	Non implementato
TS-33	Verificare che il <i>sistema_G</i> possa fornire all'amministratore la possibilità di visualizzare la lista totale degli ingredienti inclusi nella <i>prenotazione_G</i>	RDF 33	Non implementato
TS-34	Verificare che il <i>sistema_G</i> possa consentire all'amministratore di visualizzare tutti gli ordini confermati per il ristorante selezionato	ROF 34	Non implementato
TS-35	Verificare che il <i>sistema_G</i> possa permettere di annullare l' <i>ordinazione_G</i> collaborativa nel tempo utile per farlo	ROF 35	Non implementato
TS-36	Verificare che il <i>sistema_G</i> invii una notifica a tutti gli utenti associati alla <i>prenotazione_G</i> la cui <i>ordinazione_G</i> collaborativa è stata annullata	RDF 36	Non implementato
TS-37	Verificare che il <i>sistema_G</i> fornisca un'interfaccia per consentire agli utenti non autenticati di registrarsi come utenti base	ROF 37	Non implementato
TS-38	Verificare che l'utente base e l'amministratore possano inserire le proprie informazioni personali durante la registrazione come nome, cognome, email, password ed eventuali intolleranze	ROF 38	Non implementato

TS-39	Verificare che l'utente base e l'amministratore possano confermare di volersi registrare con le informazioni fornite prima di completare la registrazione	ROF 39	Non implementato
TS-40	Verificare che il <i>sistema_G</i> gestisca correttamente gli errori nell'inserimento delle informazioni durante la registrazione	ROF 40	Non implementato
TS-41	Verificare che l'utente base e l'amministratore possano visualizzare il menu' del ristorante selezionato	ROF 41	Non implementato
TS-42	Verificare che l'amministratore possa modificare le informazioni del proprio ristorante (nome, indirizzo, orari, coperti e tipologia di cucina)	ROF 42	Non implementato
TS-43	Verificare che la modifica delle informazioni del ristorante da parte dell'amministratore non causi problemi di sincronizzazione nella visualizzazione, ricerca del ristorante e nell' <i>ordinazione_G</i> da parte dell'utente base	RDF 43	Non implementato
TS-44	Verificare che l'utente autenticato possa visualizzare una lista con le prenotazioni passate e future	ROF 44	Non implementato
TS-45	Verificare che l'utente autenticato possa rilasciare una recensione (con anche una votazione di gradimento tramite stelle) sui ristoranti nei quali ha effettuato almeno una <i>prenotazione_G</i>	ROF 45	Non implementato
TS-46	Verificare che l'utente autenticato possa visualizzare gli ordini di un tavolo	ROF 46	Non implementato
TS-47	Verificare che l'utente autenticato possa visualizzare le recensioni rilasciate ed eventualmente eliminarle	ROF 47	Non implementato
TS-48	Verificare che l'utente generico possa visualizzare le recensioni di un ristorante e visualizzare per ognuna di essa le relative informazioni	ROF 48	Non implementato
TS-49	Verificare che l'amministratore possa inserire le informazioni del ristorante di cui è amministratore	ROF 49	Non implementato
TS-50	Verificare che l'utente/l'amministratore possa inserire la propria email e la password durante la fase di login	ROF 50	Non implementato
TS-51	Verificare che il <i>sistema_G</i> verifichi che le informazioni inserite nel login corrispondano ad un account esistente nel <i>sistema_G</i>	ROF 51	Non implementato
TS-52	Verificare che il <i>sistema_G</i> predisponga in fase di login di un'opzione per il recupero della password	ROF 52	Non implementato
TS-53	Verificare che l'utente non autenticato possa inserire la propria email durante il <i>processo_G</i> di recupero password	ROF 53	Non implementato
TS-54	Verificare che, se l'email inserita dall'utente corrisponde a un account nel <i>sistema_G</i> , il <i>sistema_G</i> invii un'email contenente un link per il recupero della password	ROF 54	Non implementato
TS-55	Verificare che, se l'email inserita dall'utente non corrisponde a un account nel <i>sistema_G</i> , il <i>sistema_G</i> lo mostri a schermo	ROF 55	Non implementato

TS-56	Verificare che l'utente possa accedere a una sezione dedicata tramite il link fornito nell'email di recupero password	ROF 56	Non implementato
TS-57	Verificare che il <i>sistema_G</i> mostri una lista di prenotazioni, ognuna con le seguenti informazioni: Nome del ristorante, Data, Ora, Stato della <i>prenotazione_G</i> (se è ancora attiva o già completata) e Numero di persone coinvolte	ROF 57	Non implementato
TS-58	Verificare che il <i>sistema_G</i> ordini la lista delle prenotazioni per data della <i>prenotazione_G</i>	RDF 58	Non implementato
TS-59	Verificare che l'utente possa inserire le proprie allergie/intolleranze, se presenti, durante la modifica del profilo	ROF 59	Non implementato
TS-60	Verificare che l'utente autenticato e l'amministratore possano inserire nome, cognome, mail e password nell'area di modifica delle proprie informazioni	ROF 60	Non implementato
TS-61	Verificare che il <i>sistema_G</i> verifichi che l'email inserita dall'utente base/amministratore sia valida e non sia già presente nel <i>sistema_G</i>	ROF 61	Non implementato
TS-62	Verificare che l'utente base autenticato possa accedere alla funzionalità di <i>ordinazione_G</i> di un piatto	ROF 62	Non implementato
TS-63	Verificare che il <i>sistema_G</i> consenta all'utente di visualizzare, aggiungere e togliere ingredienti del piatto selezionato	ROF 63	Non implementato
TS-64	Verificare che l'amministratore possa visualizzare le recensioni del proprio ristorante	ROF 64	Non implementato
TS-65	Verificare che l'amministratore possa rispondere alle recensioni del proprio ristorante	ROF 65	Non implementato
TS-66	Verificare che l'utente possa visualizzare le risposte alla propria recensione	ROF 66	Non implementato
TS-67	Verificare che il <i>sistema_G</i> consenta di modificare l'ordine ad un utente prima della scadenza del tempo previsto, inserendo piatti, modificando ingredienti e quantità delle pietanze ordinate	ROF 67	Non implementato
TS-68	Verificare che il <i>sistema_G</i> permetta di far visualizzare all'amministratore il dettaglio degli ingredienti necessari per ogni giornata	RDF 68	Non implementato
TS-69	Verificare che il <i>sistema_G</i> consenta all'utente autenticato di selezionare l'opzione di logout	ROF 69	Non implementato
TS-70	Verificare che, dopo che l'utente ha selezionato l'opzione di logout, il <i>sistema_G</i> richieda una conferma esplicita dall'utente prima di procedere con la disconnessione	ROF 70	Non implementato
TS-71	Verificare che, dopo aver terminato la sessione dell'utente, il <i>sistema_G</i> reindirizzi l'utente alla pagina di accesso o a una pagina di destinazione predefinita	ROF 71	Non implementato

TS-72	Verificare che l'utente autenticato possa visualizzare le informazioni del suo profilo	ROF 72	Non implementato
TS-73	Verificare che l'amministratore possa visualizzare tutte le chat aperte in precedenza da altri utenti	ROF 73	Non implementato
TS-74	Verificare che l'amministratore possa rispondere alle chat con gli utenti	ROF 74	Non implementato

Tabella 11: Test di sistema

4 Checklist

La verifica tramite analisi statica è preferibile che avvenga tramite *Inspection_G* anziché *Walkthrough_G*, per questo motivo sono definite delle liste di controllo aggiornate progressivamente dai verificatori che permettono rapidamente di analizzare gli errori più comuni e di verificare selettivamente.

4.1 Documentazione

4.1.1 Struttura

Errore	Descrizione
Manca la <i>caption_G</i> per le tabelle o le immagini	Ogni tabella o immagine deve possedere una <i>caption_G</i>
Sezione vuota	Le sezioni vuote devono essere eliminate
Parola non coincide con il glossario	Tale parola deve essere riportata con le stesse lettere maiuscole e minuscole rispetto al glossario eccetto la lettera iniziale

Tabella 12: Checklist dei possibili errori nella struttura della documentazione

4.1.2 Errori ortografici

Errore	Descrizione
Errore di sintassi	Errori di battitura o di distrazione devono essere rimossi
Errore di coniugazione	Gli errori di coniugazione devono essere rimossi

Tabella 13: Checklist dei possibili errori ortografici nella documentazione

4.1.3 Analisi Dei Requisiti

Errore	Descrizione
Requisiti per un <i>caso d'uso_G</i> assenti	Ad ogni <i>caso d'uso_G</i> deve corrispondere almeno un requisito
Diagrammi dei casi d'uso erronei	I diagrammi dei casi d'uso devono riportare la corretta numerazione e nomenclatura del <i>caso d'uso_G</i> stesso e riferirsi allo stesso modo ad altri casi d'uso

Tabella 14: Checklist dei possibili errori per l'Analisi Dei Requisiti

5 Cruscotto di valutazione della qualità

5.1 MPC01-EAC (Estimated at Completion)

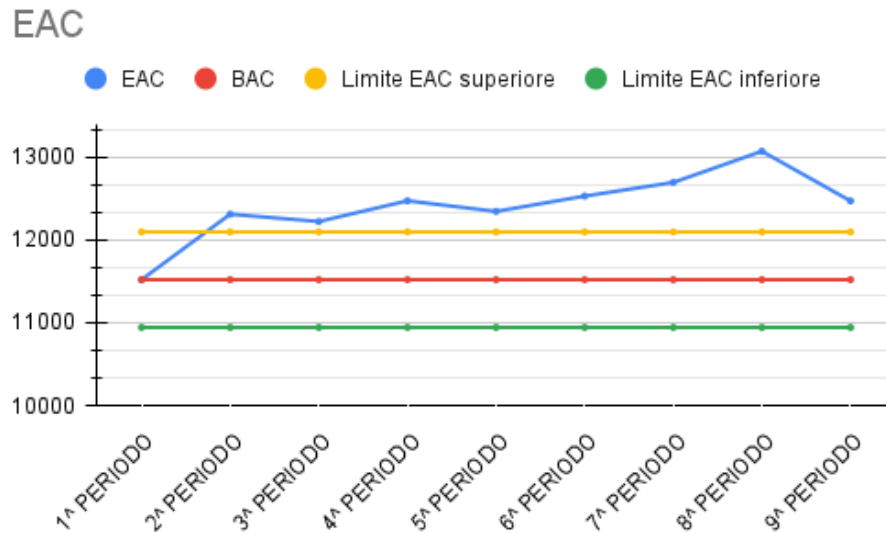


Figura 1: Estimated at Completion

Dal grafico si può notare che l'EAC supera il valore accettabile di quest'ultimo. La causa di questa situazione si può ricondurre ad una scorretta divisione tra ore produttive e individuali e, inoltre, ai lunghi periodi di tempo impiegati a imparare le *tecnologie_G* necessarie per lo svolgimento del progetto. Si può notare un andamento finale decrescente, quindi si ritiene che dopo la fase iniziale, il gruppo possa rientrare nei valori ottimali. Infine il team è consapevole che l'andamento iniziale del valore è dovuto ad una gestione errata del progetto e quindi i membri dovranno impegnarsi per rientrare nel valore ottimale e garantire il raggiungimento degli obiettivi fissati.

5.2 MPC02-PV (Planned Value) e MPC04-EV (Earned Value)

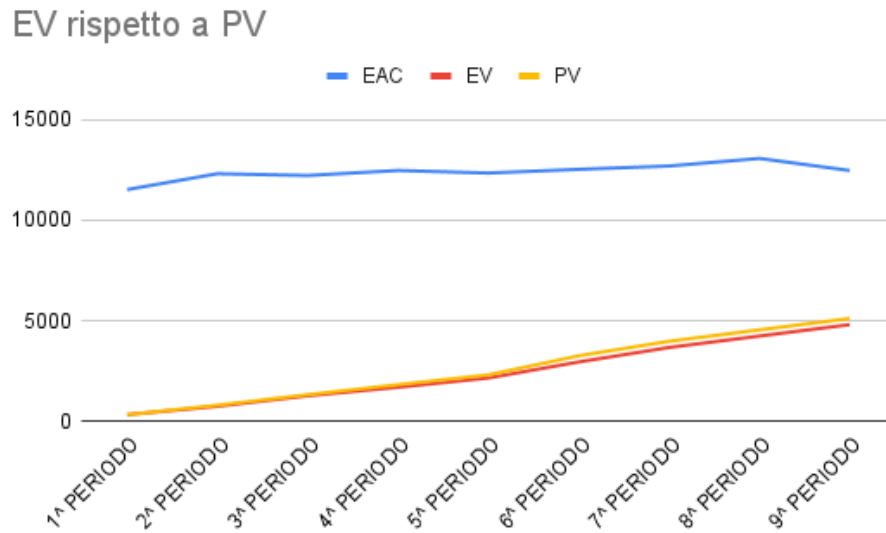


Figura 2: Planned Value - Earned Value

Dal grafico si può notare che le linee del PV e dell'EV sono molto vicine anche se si può notare un allontanamento dei due valori negli ultimi periodi. Questo denota che il costo preventivato è maggiore del costo sostenuto dal gruppo e quindi il team dovrà impegnarsi maggiormente per terminare le attività stabilite.

5.3 MPC05-ETC (Estimated to Complete)

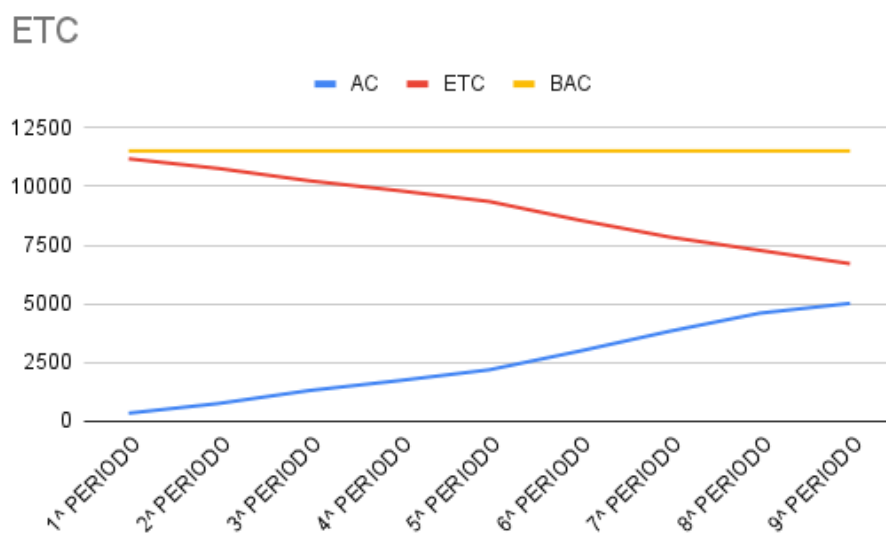


Figura 3: Estimated to Complete

Dal grafico si può notare che le linee dell'AC e dell'ETC nel corso dei vari periodi, mantengano un andamento costante. Di conseguenza si può affermare che il progetto stia mantenendo un ritmo regolare di avanzamento.

5.4 MPC06-CV (Cost Variance), MPC07-SV (Schedule Variance) e MPC08-BV (Budget Variance)

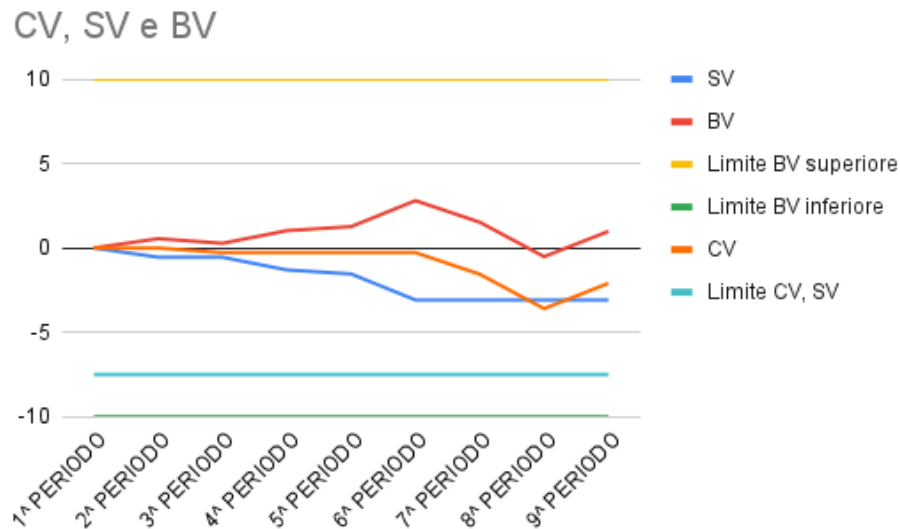


Figura 4: Cost Variance - Schedule Variance - Budget Variance

Dal grafico si può notare che la linea della Cost Variance ha mantenuto un andamento costante e lineare fino al sesto periodo, cosa positiva. Dal settimo periodo periodo però c'è stato un calo, ciò si può ricondurre alla quasi assente divisione delle ore produttive da quelle individuali. Osservando la Schedule Variance si nota invece che l'allontanamento dal valore preferibile è iniziato molto prima. Questa situazione è stata causata da un misto di mancata divisione delle ore produttive e ore individuali, e di sottostime delle ore da dare per ogni lavoro. Infine, osservando la Budget Variance, si nota che, nonostante si sia speso più di quanto previsto per la maggior parte del tempo, ciò si è stabilizzato negli ultimi periodi.

5.5 MPC15-NCR (Non Calculated Risk)

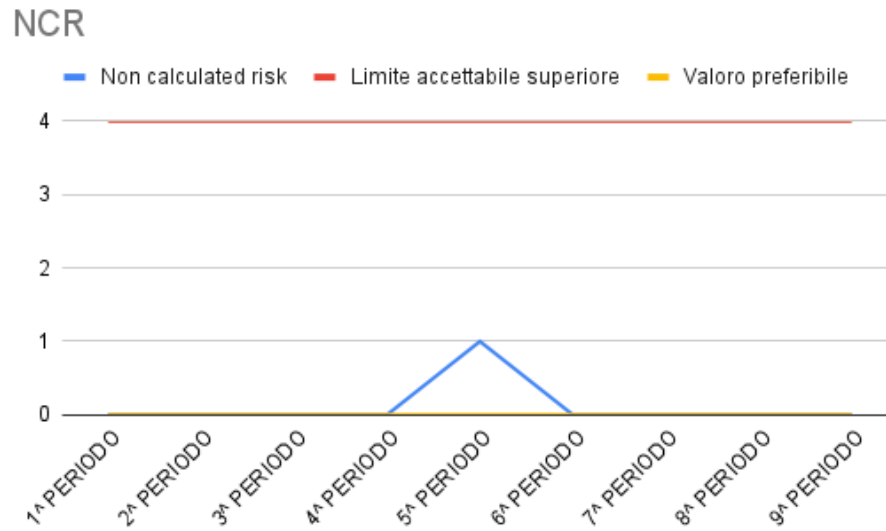


Figura 5: Non Calculated Risk

Dal grafico si può notare che per la maggior parte del tempo non sono comparsi $rischi_G$ non previsti. Solo nel quinto periodo è emerso un $rischi_G$ di cui non si era tenuto conto inizialmente, ovvero la sessione di esami, la quale ha portato via parecchio tempo ai vari membri del gruppo, rallentando di molto l'avanzamento del lavoro.

6 Valutazioni per il miglioramento

In questa sezione viene riportata la valutazione generale sul lavoro con lo scopo di inserire osservazioni sui problemi presenti e sulle possibili correzioni da adottare come miglioramenti.

6.1 Valutazione sull'organizzazione

Problema	Descrizione	Gravità	Soluzione
Tracciamento temporale	Il gruppo, nella fase iniziale del progetto, ha avuto delle difficoltà nel tracciamento delle ore per ogni attività del ruolo che copre ogni membro	Media	Adottato l'uso di <i>Jira_G</i> per suddividere anche graficamente ogni attività in base allora <i>sprint_G</i> , con relativo tracciamento temporale per ogni task.
Meeting di gruppo	Il gruppo, avendo diversi membri alle prese con esami arretrati o per esigenze lavorative, ha avuto qualvolta difficoltà nell'organizzare dei meeting di gruppo in cui fossero presenti tutti i partecipanti	Bassa	Con il passare del tempo dall'inizio del progetto si sono individuati le effettive fasce orarie in cui l'intero gruppo era disponibile per organizzare i meeting, salve imprevisti
Definizione casi d'uso	Inizialmente è stato perso un po' di tempo per via della mancata definizione di un pattern da rispettare per trascrivere i casi d'uso nell' <i>Analisi Dei Requisiti_G</i> .	Media	In gruppo si è deciso un metodo per trascrivere ogni <i>caso d'uso_G</i> , come ad esempio come devono essere fatti i diagrammi, cosa deve esserci nei vari scenari, cosa devono contenere i sottocasi e altre migliorie di carattere organizzativo.

Tabella 15: Valutazione sull'organizzazione

6.2 Valutazione sugli strumenti utilizzati

Problema	Descrizione	Gravità	Soluzione
Poca conoscenza delle nuove <i>tecnologie_G</i>	Il gruppo si è trovato a dover operare con <i>tecnologie_G</i> fortemente consigliate dal proponente ma di cui il gruppo in generale aveva poca affinità.	Media	Suddivisione della comprensione delle nuove <i>tecnologie_G</i> in base al ruolo che si sta coprendo in quel momento e integrazione delle <i>tecnologie_G</i> che non si sanno ancora grazie a chi le ha studiate e imparate in precedenza in ruoli passati (scambio di informazioni tra i membri).
Suddivisione delle task	Il gruppo si è ritrovato a assegnare delle task che però a loro volta sarebbero dovute essere assegnate a più membri.	Bassa	Creazione di sub-task in modo da assegnare allo specifico membro la sua parte di task da eseguire per completare la task padre.

Tabella 16: Valutazione sugli strumenti utilizzati

6.3 Valutazione sui ruoli

Problema	Descrizione	Gravità	Soluzione
Verifica Analisi dei Requisiti	Verificare l' <i>Analisi Dei Requisiti_G</i> si è rivelato più complesso del previsto per via di casi d'uso ridondanti o espressi male	Alta	Integrazione di 2 verificatori che simultaneamente verificavano tutto il documento per un periodo dedicato della settimana
Responsabile	Prima dell'uso di <i>Jira_G</i> , la spartizione dei compiti e della gestione delle ore era abbastanza vaga e poco tracciata	Media	Adottare <i>Jira_G</i> e ogni volta venisse completata una task, il Responsabile teneva traccia del tempo per portarla a termine per fare il resoconto dei costi e dei tempi rispetto al <i>preventivo_G</i> .

Tabella 17: Valutazione sui ruoli