



4OURSQUARED

4oursquared.unipd@gmail.com

Piano di Qualifica

Informazioni

<i>Redattori</i>	Brotto Romina
	Salami Lorenzo
	Soldà Matteo
<i>Versione</i>	1.0.2
<i>Uso</i>	esterno

Descrizione

Questo documento serve a definire le metriche e i criteri di accettazione dei prodotti.



Versione	Data	Redattore	Verificatore	Descrizione
1.0.2	19/09/2023	Brotto Romina		Aggiunti test server
1.0.1	19/09/2023	Brotto Romina	Salami Lorenzo	Aggiunta sezione di test client
1.0.0	25/07/2023	Salami Lorenzo	Brotto Romina	Versione finale per RTB
0.1.0	25/07/2023	Brotto Romina	Salami Lorenzo	Verifica per RTB
0.0.2	24/07/2023	Brotto Romina	Ceccato Francesco	Aggiunta sezione 3 ed indici
0.0.1	24/04/2023	Soldà Matteo	Salami Lorenzo	Aggiunta delle intestazioni e dei piè di pagina
0.0.0	18/04/2023	Salami Lorenzo	Soldà Matteo	Stesura iniziale.



Contents

1	Qualità di prodotto	1
1.1	Documentazione	1
1.1.1	Indice di Gulpease	1
1.2	Prodotti software	1
1.2.1	Copertura statement	1
1.2.2	Copertura branch	1
2	Qualità di processo	2
2.0.1	Time variance	2
2.0.2	Budget variance	2
3	Applicazione e valutazione delle metriche	2
3.1	Valutazione d'insieme (Qualità di processo)	2
4	Attività di Testing	2
5	Metriche	9
5.1	Cost Performance Index	10

1 Qualità di prodotto

1.1 Documentazione

1.1.1 Indice di Gulpease

$$\text{Indice di Gulpease} = 89 + \frac{300 * \# \text{frasi} - 10 * \# \text{lettere}}{\# \text{parole}}$$

- #lettere: numero di caratteri alfanumerici;
- #parole: numero di gruppi di caratteri alfanumerici;
- #frasi: numero di gruppi di punti o punti e virgola consecutivi.

Prodotti coinvolti:

Prodotto	Valore accettabile	Valore ottimale
Documenti interni	> 40	> 60
Documenti esterni	> 50	> 60

Riferimenti: <http://www.corrige.it/leggibilita/lindice-gulpease/>

1.2 Prodotti software

1.2.1 Copertura statement

La metrica si basa sullo statement coverage.

Prodotti coinvolti:

Prodotto	Valore accettabile	Valore ottimale
Software	> 80%	> 95%

1.2.2 Copertura branch

La metrica si basa sul branch coverage.

Prodotti coinvolti:

Prodotto	Valore accettabile	Valore ottimale
Software	> 55%	> 80%

2 Qualità di processo

2.0.1 Time variance

La metrica si basa sulla variazione percentuale rispetto alla stima iniziale.

Prodotti coinvolti:

Prodotto	Valore accettabile	Valore ottimale
Software	< 20%	0%
Documentazione	< 20%	0%

2.0.2 Budget variance

La metrica si basa sulla variazione percentuale rispetto alla stima iniziale.

Prodotti coinvolti:

Prodotto	Valore accettabile	Valore ottimale
Software	< 20%	0%
Documentazione	< 20%	0%

3 Applicazione e valutazione delle metriche

I grafici sono frutto di un foglio di calcolo creato dal gruppo che applica le formule per il calcolo delle metriche definite in questo documento.

3.1 Valutazione d'insieme (Qualità di processo)

L'avanzamento del lavoro è proseguito secondo le aspettative. È stato riscontrato una diminuzione delle ore lavorate negli sprint 8 e 9 dovuti a impegni universitari quali esami e consegne. Questo calo ha particolarmente influenzato il grafico dello schedule variance che è effettivamente sceso sotto la soglia di tolleranza prefissata dal gruppo. Lo stesso si può riscontrare nella distanza tra il planned e l'earned value, che è cresciuta particolarmente durante quegli sprint.

Il gruppo aveva tenuto in considerazione che ci sarebbe stato un calo di lavoro durante gli sprint sopra indicati, prevedendo però di rientrare nei valori di tolleranza negli sprint successivi.

4 Attività di Testing

I seguenti test sono stati elencati seguendo l'ordine alfabetico dei nomi dei file in cui si trovano.



Test Client

Codice	Descrizione	Requisiti collegati
LoginPage.test.tsx file		
TC1	Verifica che il client renderizzi correttamente la pagina di login.	RF5-O, RV1-O
TC2	Verifica che le credenziali inserite siano corrette e che al click del pulsante "Entra" il login abbia successo	RF5-O, RV1-O
TC3	Verifica che le credenziali inserite non siano corrette e che al click del pulsante "Entra" il login <i>non</i> abbia successo	RF5-O, RV1-O
AreaSingleView.test.tsx file		
TC4	Verifica che il fetch delle informazioni riguardanti tutte le aree illuminate avvenga correttamente	RF7-O
TC5	Verifica che, dopo la cancellazione del lampione, esso sia effettivamente scomparso dalla lista	RF20-O
TC6	Verifica che, dopo la cancellazione del sensore, esso sia effettivamente scomparso dalla lista	RF18-O
AreaTable.test.tsx file		
TC7	Verifica che il fetch delle informazioni per la singola area avvenga correttamente	RF10-O
TC8	Verifica che la riconfigurazione dell'area illuminata avvenga con successo	RF11-O
TC9	Verifica che la rimozione dell'area illuminata avvenga con successo	RF12-O
TC10	Verifica che la rimozione dell'area illuminata venga annullata senza modificarne il contenuto	Rf12-O



Content.test.tsx file		
TC11	Verifica che la pagina di elenco aree illuminate si carichi correttamente (anche se vuota)	RF7-O
EditAreaForm.test.tsx file		
TC12	Verifica che il form di modifica area illuminata venga visualizzato correttamente	RF11-O
Footer.test.tsx file		
TC13	Verifica che il footer comune a tutte le pagine venga visualizzato correttamente	-
Header.test.tsx file		
TC14	Verifica che il render del modulo header per il logout e il footer avvengano correttamente	-
LampCard.test.tsx file		
TC16	Verifica che il render della scheda dettagli lampione sia visualizzata correttamente	RF19-O
LampGuastiPage.test.tsx file		
T16	Verifica che il render della pagina dedicata all'elenco aree illuminate segnalate guaste avvenga correttamente	RF8-O, RV2-O
LampGuastiTable.test.tsx file		
TC17	Verifica che il load della tabella per l'elenco delle aree illuminate segnalate guaste avvenga correttamente	RF8-O, RV2-O
TC18	Verifica che la rimozione dell'area illuminata dall'elenco delle aree guaste avvenga con successo	RF16-O
TC19	Verifica che la rimozione dell'area illuminata dall'elenco delle aree guaste venga annullata	RF12-O
LampSingleView.test.tsx file		
TC20	Verifica che il caricamento dei dati relativi al lampione avvenga correttamente	RF11-O



PageFullView.test.tsx file		
TC21	Verifica se sono presenti header, main e footer in tutte le pagine	-
SensorSingleView.test.tsx file		
TC22	Verifica che il caricamento dei dati relativi al sensore avvenga correttamente	RF39-O
TC23	Verifica che il sensore invii correttamente il segnale per scatenare la modifica dell'intensità luminosa del lampione	RF27-O,RF14-O



Test Server

Codice	Descrizione
AccountRoutes.test.ts file	
TS1	Verifica che il meccanismo di login funzioni correttamente: ritorna 401 se l'username non è presente, 401 se la password è errata.
TS2	Verifica che il meccanismo di logout funzioni correttamente: ritorna 403 se non è stato passato un token valido, ritorna 200 se è stato passato un token valido.
TS3	Verifica che la registrazione di un nuovo utente avvenga correttamente: ritorna 200 e crea un nuovo utente con successo, ritorna errore 409 se l'username esiste già, ritorna errore 409 se la mail esiste già.
TS4	Verifica il passaggio dei token: ritorna 403 se non è stato passato un token valido, ritorna 200 se è stato passato un token valido.
TS5	Verifica il recupero della lista utenti: lista completa e/o utente specifico se l'utente è admin, ritorna 500 se l'utente non è autorizzato.
TS6	Verifica che l'aggiornamento dati utente avvenga correttamente: ritorna 404 se l'utente non è presente, 200 se l'utente è stato aggiornato.
AreaRoutes.test.ts file	
TS7	Verifica che i dati dell'area illuminata vengano ricevuti correttamente in modalità GET: ritorna 500 se non è possibile recuperare le aree, ritorna 200 e l'area in base all'id fornito, ritorna 404 se l'area non è presente, 500 se non è possibile recuperare l'area specificata dall'id fornito.
TS8	Verifica il funzionamento della creazione di un'area illuminata in modalità POST: ritorna 200 se l'area viene creata correttamente, 500 altrimenti.
TS9	Verifica che l'aggiornamento di un'area in modalità PUT avvenga correttamente: ritorna 200 se l'area viene aggiornata correttamente, 404 se l'area non viene trovata, 500 se l'area non viene aggiornata correttamente.



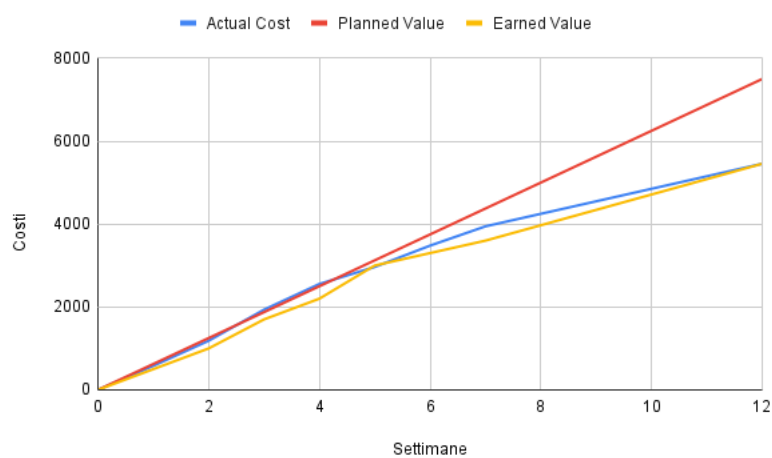
TS10	Verifica l'eliminazione di un'area: ritorna 200 se l'area viene eliminata correttamente, 404 l'area da eliminare non viene trovata, 500 se avviene un errore.
LampRoutes.test.ts file	
TS11	Verifica del recupero delle info del singolo lampione in modalità GET: ritorna 200 nel recupero del singolo lampione, 404 nel recupero di un lampione non esistente e di un lampione di un'area non esistente, 500 nel caso di un errore durante il recupero.
TS12	Verifica del recupero delle info di tutti i lampioni in modalità GET: ritorna 200 se recupera correttamente tutti i lampioni, 404 se non viene trovata l'area di riferimento, 500 se avviene un errore durante il recupero.
TS13	Verifica per l'inserimento di un lampione in modalità POST: ritorna 200 se il lampione viene aggiunto, 500 in caso di errore durante l'inserimento.
TS14	Verifica per la modifica di un lampione in modalità PUT: ritorna 200 se il lampione non guasto viene segnalato come guasto, 404 se non è presente il lampione o se non esiste l'area, 500 se avviene un errore durante la modifica.
TS15	Verifica per la richiesta di supporto alla modifica della luminosità dei lampioni: ritorna 200 se la richiesta di supporto è stata effettuata correttamente, 404 se non è presente l'area, 500 se avviene un errore.
TS16	Verifica per la cancellazione di un lampione: ritorna 200 se il lampione viene cancellato, 404 se non è presente l'area con il lampione da rimuovere, 500 in caso di errore.
TS17	Verifica per i lampioni guasti con utente manutentore: ritorna 200 se il lampione viene tolto dai guasti, 409 se il lampione non è presente, 500 se avviene un errore.
TS18	Verifica per il recupero dei lampioni guasti: ritorna 200 se recupera correttamente tutti i lampioni guasti, 404 se non viene trovata l'area specificata.



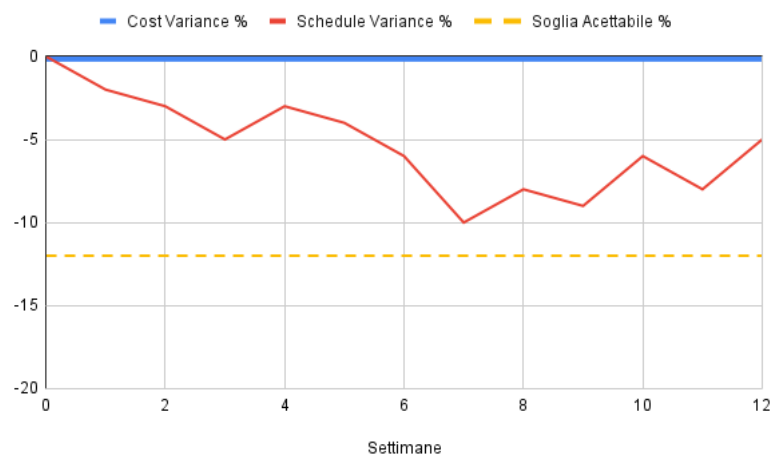
SensorRoutes.test.ts file	
TS19	Verifica del recupero delle informazioni dei sensori in modalità GET: se l'area esiste ritorna 200 e la lista di tutti i sensori presenti, ritorna 404 se l'area non esiste, 500 se non è possibile recuperare l'area specificata dall'id.
TS20	Verifica del recupero delle info del singolo sensore in modalità GET: ritorna 200 se area e sensore richiesti esistono, 404 se l'area esiste ma non il sensore o l'area non esiste
TS21	Verifica per l'inserimento di un sensore in modalità POST: ritorna 200 se l'area esiste, 400 se l'area non esiste, 500 se avviene un errore nel processo di creazione del sensore.
TS22	Verifica per la modifica di un sensore in modalità PUT: ritorna 200 se il sensore che si cerca di modificare esiste, 404 se non esiste
TS23	Verifica per la cancellazione di un sensore: ritorna 200 se il sensore che si cerca di eliminare esiste, 404 se il sensore appartiene a un'area inesistente, 500 per un errore durante il processo.
SignalRoutes.test.ts file	
TS24	Verifica funzionamento accensione e spegnimento in modalità PUSH: ritorna 400 se l'area non è stata trovata, ritorna 200 se esiste l'area e i lampioni vengono accesi, 500 in caso di errore
TS25	Verifica il post di un segnale: ritorna 200 ed aggiunge il segnale
TS26	Verifica la generazione di un token: ritorna 200 se il token viene generato correttamente.
TS27	Verifica di un token: ritorna 226 se il token è stato utilizzato, 218 se il token è valido e non è stato utilizzato, 200 se il token è valido ed è stato usato ma vanno spenti i lampioni, 498 se il token non è stato trovato, 400 se l'area non è stata trovata.

5 Metriche

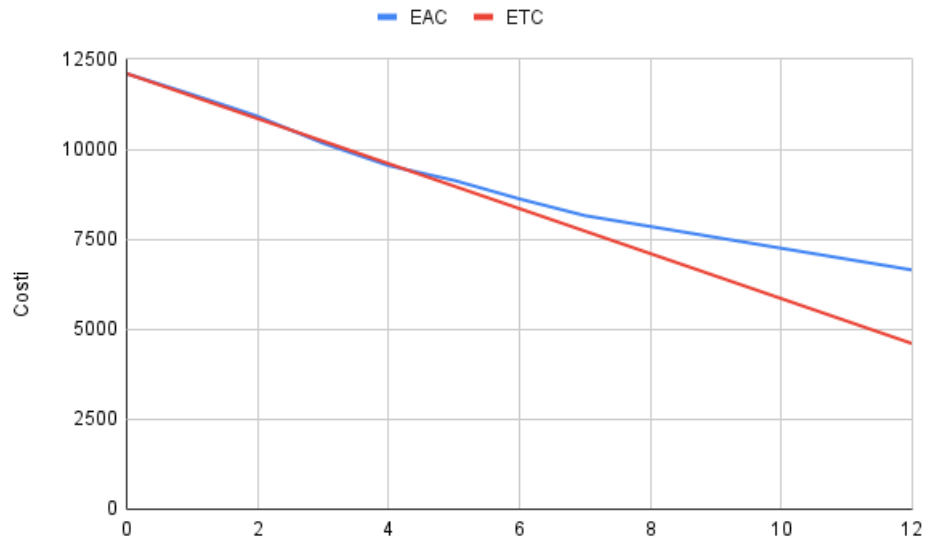
Planning Value, Actual Cost e Earned Value



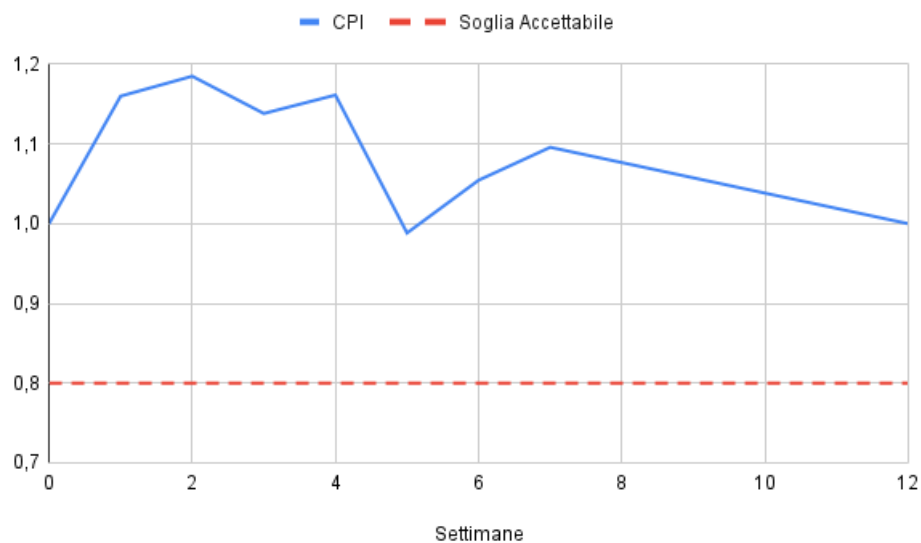
Cost Variance e Schedule Variance



Estimate at completion e Estimate to Complete



5.1 Cost Performance Index



Indice di Gulpease

I valori riportati sono frutto di un'analisi approssimativa, usando la libreria PyPdf2 per Python 3.



Documento	Frase	Parole	Caratteri	Indice
Analisi dei requisiti	839	5642	34407	73
Piano di Progetto	253	1317	7273	91
Piano di Qualifica	43	179	1128	98
Norme di Progetto	423	3228	17766	73
Glossario	131	984	5096	77