



Piano di Qualifica Progetto ShopChain

yakuzaisi.swe@gmail.com

Informazioni sul documento

Responsabile	Matteo Midena
Redattori	Francesco Bugno Luca Carturan
Verificatori	Michele Filosofo
Uso	Esterno
Destinatari	Prof. Tullio Vardanega Prof. Riccardo Cardin Ing. Fabio Pallaro - Sync Lab S.r.l.
Versione	1.0.0

Sommario

Questo documento raccoglie le strategie di qualifica e gli obiettivi di qualità perseguiti dal gruppo *Yakuzaisi* nello svolgimento del progetto ShopChain

Registro delle Modifiche

Versione	Data	Autore	Ruolo	Descrizione
1.0.0	2022/02/21	Matteo Midena	Responsabile	Approvato per il rilascio
0.4.0	2022/02/20	Michele Filosofo	Verificatore	Verifica generale documento
0.3.2	2022/02/10	Francesco Bugno	Verificatore	Sistemi grafici §A
0.3.1	2022/02/04	Luca Carturan	Verificatore	Stesura §A
0.3.0	2022/01/25	Michele Filosofo	Verificatore	Verifica generale documento
0.2.4	2022/01/22	Francesco Bugno	Progettista	Correzione errori ortografici, aggiornata §4
0.2.3	2022/01/20	Luca Carturan	Progettista	Terminata stesura §4
0.2.2	2022/01/18	Francesco Bugno	Progettista	Aggiornamento §3, iniziata stesura §4
0.2.1	2022/01/18	Luca Carturan	Progettista	Continuo stesura §3, aggiornate tabelle
0.2.0	2022/01/17	Michele Filosofo	Verificatore	Verifica generale del documento
0.1.3	2022/01/14	Francesco Bugno	Amministratore	Correzione errori ortografici, iniziata stesura §3
0.1.2	2022/01/09	Luca Carturan	Amministratore	Aggiornamento tabella §2, stesura §2.2.2 e §2.2.3
0.1.1	2022/01/04	Francesco Bugno	Amministratore	Iniziata stesura §2
0.1.0	2022/01/06	Michele Filosofo	Verificatore	Verifica generale del documento
0.0.2	2021/12/27	Luca Carturan	Amministratore	Stesura §1
0.0.1	2021/12/26	Luca Carturan	Amministratore	Creata struttura del documento

Contenuti

1	Introduzione	1
1.1	Scopo del documento	1
1.2	Scopo del capitolato	1
1.3	Glossario	1
1.4	Riferimenti	1
1.4.1	Riferimenti normativi	1
1.4.2	Riferimenti informativi	1
2	Qualità di processo	3
2.1	Obiettivi di qualità di processo	3
2.2	Metriche utilizzate	4
2.2.1	Processi primari	4
2.2.2	Processi di supporto	4
2.2.3	Processi organizzativi	4
3	Qualità di prodotto	5
3.1	Metriche utilizzate	6
4	Specifica dei test	7
4.1	Test di accettazione	7
4.2	Test di sistema	7
4.3	Test d'integrazione	9
4.4	Test di unità	9
4.4.1	Tracciamento test di unità - Solidity	12
A	Resoconto attività di verifica	14
A.1	Verifica dei documenti	14
A.1.1	Indice di Gulpease	14
A.1.2	Errori ortografici	14
A.2	Verifica dei processi	15
A.2.1	Estimated At Completion	15
A.2.2	Earned Value & Planned Value	15
A.2.3	Actual Cost & Estimate To Complete	15
A.2.4	Schedule Variance & Cost Variance	16
A.2.5	Requirements Stability Index	16
A.2.6	Attualizzazione dei rischi	16
A.2.7	Metrics Satisfied	17
A.3	Verifica dei software	17
A.3.1	Facilità di utilizzo	17
A.3.2	Versioni browser supportate	17
A.3.3	Copertura requisiti obbligatori	18
A.3.4	Copertura requisiti desiderabili	18
A.3.5	Copertura requisiti opzionali	18
A.3.6	Solidity Statement Coverage	19
A.3.7	Solidity Branch Coverage	19
A.3.8	Solidity Function Coverage	19
A.3.9	Solidity Line Coverage	20

Elenco delle tabelle

2	Obiettivi di qualità di processo	3
3	Metriche di qualità dei processi primari	4
4	Metriche di qualità dei processi di supporto	4
5	Metriche di qualità dei processi organizzativi	4
6	Obiettivi di qualità di prodotto	5
7	Metriche di qualità di prodotto	6
8	Test di sistema	9
9	Test d'integrazione	9
10	Test di unità - Solidity	11
11	Tracciamento test di unità - metodi Solidity	13

Elenco delle figure

1	Indice di Gulpease di ciascun documento per periodo	14
2	Errori ortografici individuati per periodo	14
3	Revisione del valore stimato per la realizzazione del progetto	15
4	Valore delle attività realizzate e costo pianificato per realizzare le rimanenti	15
5	Costo effettivamente sostenuto e valore stimato per la realizzazione delle rimanenti attività	15
6	Cost Variance e Schedule Variance per periodo	16
7	Variazione del numero di requisiti	16
8	Rischi verificati per periodo	16
9	Percentuale di metriche soddisfatte per periodo	17
10	Quantitativo di click necessari per raggiungere la funzione desiderata	17
11	Versioni di browser supportate per periodo	17
12	Percentuale di requisiti obbligatori implementati per periodo	18
13	Percentuale di requisiti desiderabili implementati per periodo	18
14	Percentuale di requisiti opzionali implementati per periodo	18
15	Statement Coverage riguardante il codice Solidity	19
16	Branch Coverage riguardante il codice Solidity	19
17	Function Coverage riguardante il codice Solidity	19
18	Line Coverage riguardante il codice Solidity	20



1 Introduzione

1.1 Scopo del documento

Questo documento ha lo scopo di descrivere le strategie che il gruppo usa per perseguire gli obiettivi di qualità da applicare al progetto.

Esso deve implementare degli standard che permettano il miglioramento continuo, tramite misurazioni periodiche dei risultati ottenuti, sfruttandoli per ottenere azioni migliorative.

All'interno del *Piano di progetto* vengono raccolte le definizioni dei test, il loro stato e il loro tracciamento rispetto ai requisiti individuati nell'*Analisi dei requisiti v1.0.0*.

1.2 Scopo del capitolato

L'avvento delle tecnologie BlockChain_G ha portato e porterà nei prossimi anni a grandi cambiamenti nella società.

In particolare, ha aperto le porte a una nuova forma di finanza, la cosiddetta “DeFi” (Finanza Decentralizzata) che ha permesso a chiunque sia dotato di connessione internet di creare un Wallet_G e possedere quindi criptovalute_G.

Questo ha delineato due profili critici strettamente legati: da un lato il controllo del proprio portafoglio è passato completamente nelle mani dell'utente, dall'altro lato questo comporta la mancanza di un ente terzo che si occupi di gestire transazioni e offrire garanzie.

Nel capitolato in questione si vuole proprio risolvere questo problema, in uno scenario che comprende un e-commerce_G basato su BlockChain_G in cui si vuole tutelare entrambe le parti coinvolte in un acquisto tramite criptovalute.

Il fine del progetto è la realizzazione di un prototipo di una piattaforma integrabile con un “crypto-commerce”_G, che si occupi di gestire gli ordini dalle fasi di pagamento alla consegna.

1.3 Glossario

I termini utilizzati in questo documento potrebbero generare dubbi riguardo al loro significato, richiedendo pertanto una definizione al fine di evitare ambiguità. Tali termini vengono contrassegnati da una G maiuscola finale a pedice della parola. La loro spiegazione è riportata nel *Glossario v1.0.0*.

1.4 Riferimenti

1.4.1 Riferimenti normativi

- *Norme di progetto v1.0.0*;
- **Regolamento del progetto didattico:**
<https://www.math.unipd.it/~tullio/IS-1/2021/Dispense/PD2.pdf>

1.4.2 Riferimenti informativi

- *Analisi dei requisiti v1.0.0*;
- **Capitolato d'appalto C2 - ShopChain:**
<https://www.math.unipd.it/~tullio/IS-1/2021/Progetto/C2.pdf>
- **Software Engineering - Ian Sommerville - 10th Edition:**
 - Capitolo 24 - Quality management.
- **Qualità di prodotto - slide T12 del corso di Ingegneria del Software:**
<https://www.math.unipd.it/~tullio/IS-1/2021/Dispense/T12.pdf>



- **Qualità di processo** - slide T13 del corso Ingegneria del Software:
<https://www.math.unipd.it/~tullio/IS-1/2021/Dispense/T13.pdf>
- **Verifica e validazione: introduzione** - slide T14 del corso Ingegneria del Software:
<https://www.math.unipd.it/~tullio/IS-1/2021/Dispense/T14.pdf>
- **Verifica e validazione: analisi statica** - slide T15 del corso Ingegneria del Software:
<https://www.math.unipd.it/~tullio/IS-1/2021/Dispense/T15.pdf>
- **Verifica e validazione: analisi dinamica** - slide T16 del corso Ingegneria del Software:
<https://www.math.unipd.it/~tullio/IS-1/2021/Dispense/T16.pdf>
- **ISO/IEC 9126:**
http://www.colonese.it/00-Manuali_Pubblicatii/07-ISO-IEC9126_v2.pdf
- **ISO/IEC 12207:1997:**
https://www.math.unipd.it/~tullio/IS-1/2009/Approfondimenti/ISO_12207-1995.pdf
- **Indice di Gulpease:**
https://it.wikipedia.org/wiki/Indice_Gulpease

2 Qualità di processo

Per garantire la qualità dei processi, è stato scelto di utilizzare come riferimento lo standard [ISO/IEC 12207:1997](#), dal quale sono stati semplificati e adattati alle necessità alcuni dei processi in esso elencati. In questa sezione vengono esposti i valori ottimali e accettabili riguardanti le metriche scelte, per consultare nel dettaglio le metriche sotto riportate, fare riferimento al documento *Norme di progetto v1.0.0*.

2.1 Obiettivi di qualità di processo

Obiettivo	Descrizione	Metriche
Processi primari		
Fornitura	Processo che consiste nel decidere procedure e risorse adatte a soddisfare le necessità del cliente.	MPC01 MPC02 MPC03 MPC04 MPC05 MPC06 MPC07
Sviluppo	Processo che ha lo scopo di sviluppare un prodotto software che indirizzi le esigenze del cliente.	MPC08
Processi di supporto		
Verifica	Processo che ha lo scopo di confermare che ciascun servizio realizzato soddisfi i requisiti specificati.	MPC09
Gestione della qualità	Processo che ha lo scopo di assicurare che il prodotto e i servizi offerti siano conformi agli standard definiti.	MPC10
Processi organizzativi		
Gestione organizzativa	Processo che si occupa di esporre le modalità di coordinamento del gruppo.	MPC11

Tabella 2: Obiettivi di qualità di processo

2.2 Metriche utilizzate

2.2.1 Processi primari

Codice	Nome metrica	Valore accettabile	Valore ottimale
Fornitura			
MPC01	Earned Value (EV)	> 0	$\leq EAC$
MPC02	Actual Cost (AC)	≥ 0	$\leq EAC$
MPC03	Planned Value (PV)	≥ 0	$\leq BAC$
MPC04	Cost Variance (CV)	$\geq -10\%$	$\geq 0\%$
MPC05	Schedule Variance (SV)	$\geq -10\%$	$\geq 0\%$
MPC06	Estimated At Completion (EAC)	$\geq BAC - 3\%; \leq BAC + 3\%$	$= BAC$
MPC07	Estimate To Complete (ETC)	≥ 0	$\leq EAC$
Sviluppo			
MPC08	Requirements Stability Index (RSI)	$\geq 80\%$	100%

Tabella 3: Metriche di qualità dei processi primari

2.2.2 Processi di supporto

Codice	Nome metrica	Valore accettabile	Valore ottimale
Verifica			
MPC09	Passed Tests	$\geq 80\%$	100%
Gestione della qualità			
MPC10	Metrics Satisfied	$\geq 85\%$	100%

Tabella 4: Metriche di qualità dei processi di supporto

2.2.3 Processi organizzativi

Codice	Nome metrica	Valore accettabile	Valore ottimale
Gestione organizzativa			
MPC11	Risks Found	≤ 5	0

Tabella 5: Metriche di qualità dei processi organizzativi

3 Qualità di prodotto

Per garantire la qualità di prodotto, è stato scelto di utilizzare come riferimento lo standard ISO/IEC 9126. In questa sezione vengono esposti i valori ottimali e accettabili riguardanti le metriche selezionate dal gruppo *Yakuzaishi*. Per consultare nel dettaglio le metriche sotto riportate, fare riferimento al documento *Norme di progetto v1.0.0*.

Obiettivo	Descrizione	Metriche
Monitoraggio documentazione		
Leggibilità documenti	I documenti devono essere comprensibili agli utenti.	MPD01
Correttezza linguistica	Tutti gli errori grammaticali devono essere corretti.	MPD02
Monitoraggio software		
Funzionalità	Capacità del prodotto di offrire tutte le funzioni individuate nell' <i>Analisi dei requisiti</i> , perseguendo accuratezza e adeguatezza.	MPD03 MPD04 MPD05
Usabilità	Capacità di essere comprensibile in modo da rendere piacevole l'esperienza dell'utente. Le funzionalità devono essere compatibili con le aspettative.	MPD06
Portabilità	Capacità di poter funzionare in diversi ambienti di esecuzione. Gli obiettivi da perseguire sono adattabilità e sostituibilità.	MPD07
Copertura test	Il codice prodotto dovrà essere verificato nella sua interezza al fine di garantire la corretta implementazione dei requisiti individuati.	MPD08 MPD09 MPD10 MPD11

Tabella 6: Obiettivi di qualità di prodotto

3.1 Metriche utilizzate

Codice	Nome Metrica	Valore accettabile	Valore ottimale
Documenti			
MPD01	Indice di Gulpease	$\geq 60\%$	$\geq 80\%$
MPD02	Errori ortografici	0	0
Software			
MPD03	Copertura requisiti obbligatori	100%	100%
MPD04	Copertura requisiti desiderabili	$\geq 90\%$	100%
MPD05	Copertura requisiti opzionali	$\geq 80\%$	100%
MPD06	Facilità di utilizzo	5 click	4 click
MPD07	Versioni browser supportate	$\geq 80\%$	100%
MPD08	Solidity Statement Coverage	$\geq 80\%$	100%
MPD09	Solidity Branche Coverage	$\geq 80\%$	100%
MPD10	Solidity Function Coverage	$\geq 80\%$	100%
MPD11	Solidity Line Coverage	$\geq 80\%$	100%

Tabella 7: Metriche di qualità di prodotto

4 Specifica dei test

4.1 Test di accettazione

I test di accettazione sono necessari per dimostrare che il prodotto soddisfa i requisiti minimi concordati con il proponente.

Essi si compongono dei test di sistema e vengono eseguiti durante il collaudo finale sia dai membri del gruppo che dall'azienda proponente, sotto supervisione del gruppo stesso.

4.2 Test di sistema

ID Test	Descrizione	ID Requisiti	Stato
TS1F1	Verificare che la richiesta di checkout da un E-Commerce avvenga correttamente.	R1F1	NI
TS1F2	Verificare che l'utente visualizzi correttamente le diverse tipologie di pagamento.	R1F2	NI
TS1F2.1	Verificare che l'utente possa scegliere la tipologia di pagamento unico correttamente.	R1F2.1	NI
TS1F2.2	Verificare che l'utente possa scegliere la tipologia di pagamento MoneyBox correttamente.	R1F2.2	NI
TS2F2.2.1	Verificare che l'utente possa visualizzare lo stato di completamento della MoneyBox correttamente.	R2F2.2.1	NI
TS2F2.2.2	Verificare che l'utente possa visualizzare l'invito di partecipazione alla MoneyBox correttamente.	R2F2.2.2	NI
TS3F2.2.3	Verificare che l'utente possa visualizzare una traduzione visiva della MoneyBox correttamente.	R3F2.2.3	NI
TS2F2.2.4	Verificare che in caso di chiusura della MoneyBox i soldi vengano restituiti correttamente.	R2F2.2.4	NI
TS2F2.2.5	Verificare che l'utente possa visualizzare l'elenco delle transazioni dei partecipanti alla MoneyBox correttamente.	R2F2.2.5	NI
TS1F3	Verificare che l'utente possa visualizzare il totale dell'ordine correttamente.	R1F3	NI
TS1F4	Verificare che l'utente possa effettuare la connessione a Metamask correttamente.	R1F4	NI

Continua nella pagina successiva

ID Test	Descrizione	ID Requisiti	Stato
TS1F5	Verificare che l'utente possa pagare correttamente.	R1F5	NI
TS1F5.1	Verificare che il versamento per il pagamento unico copra l'intera somma richiesta.	R1F5.1	NI
TS2F5.2	Verificare che il versamento per il pagamento tramite MoneyBox possa coprire solo una parte della somma richiesta.	R2F5.2	NI
TS1F6	Verificare che la richiesta di rimborso funzioni correttamente.	R1F6	NI
TS1F7	Verificare che l'acquirente possa sbloccare, correttamente, i fondi dallo Smart Contract dopo avvenuta ricezione.	R1F7	NI
TS1F8	Verificare che l'utente possa visualizzare le transazioni.	R1F8	NI
TS2F8.1	Verificare che il venditore possa visualizzare le transazioni in entrata pagate.	R2F8.1	NI
TS2F8.1.1	Verificare che il venditore possa visualizzare le transazioni in entrata pagate ma non sbloccate.	R2F8.1.1	NI
TS2F8.1.2	Verificare che il venditore possa visualizzare le transazioni in entrata pagate e sbloccate.	R2F8.1.2	NI
TS2F8.1.3	Verificare che il venditore possa visualizzare le transazioni in entrata pagate cancellate.	R2F8.1.3	NI
TS1F8.2	Verificare che il proprietario dell'ordine possa visualizzare le transazioni in uscita.	R1F8.2	NI
TS2F8.2.1	Verificare che il proprietario dell'ordine possa visualizzare le transazioni in uscita non pagate.	R2F8.2.1	NI
TS2F8.2.2	Verificare che il proprietario dell'ordine possa visualizzare le transazioni in uscita pagate ma non sbloccate.	R2F8.2.2	NI
TS2F8.2.3	Verificare che il proprietario dell'ordine possa visualizzare le transazioni in uscita pagate e sbloccate.	R2F8.2.3	NI
TS2F8.2.4	Verificare che il proprietario dell'ordine possa visualizzare le transazioni in uscita cancellate.	R2F8.2.4	NI

Continua nella pagina successiva

ID Test	Descrizione	ID Requisiti	Stato
TS2F9	Verificare che si possa convertire correttamente l'ammontare depositato in stable coin.	R2F9	NI
TS3F10	Verificare che, dopo la transazione, una fee percentuale venga destinata al fondo ShopChain correttamente.	R3F10	NI
TS1F11	Verificare che la richiesta di un ordine venga caricata correttamente su una blockchain pubblica.	R1V1	NI
TS1F12	Verificare che il pagamento e le transazioni siano gestite correttamente dallo smart contract.	R1V2	NI

Tabella 8: Test di sistema

4.3 Test d'integrazione

Codice	Descrizione	Stato
TI1	Verificare che il collegamento con Metamask avvenga correttamente.	NI
TI2	Verificare che il collegamento tra Front-End _G e Smart Contract tramite libreria Web3.js avvenga correttamente.	NI

Tabella 9: Test d'integrazione

4.4 Test di unità

Per garantire il corretto funzionamento di ogni minimo componente autonomo del sistema vengono eseguiti i test di unità.

Fino a questo momento sono stati presi in considerazione esclusivamente i test eseguiti su linguaggio Solidity_G. I test di unità rimanenti verranno stabiliti durante il periodo di *Progettazione di dettaglio e codifica requisiti obbligatori*.

ID Test	Descrizione	Stato
TU01	Si verifica che il contatore degli ordini sia inizializzato a zero.	S
TU02	Si verifica che l'ordine creato sia corretto.	S
TU03	Si verifica che il contratto abbia ricevuto l'ammontare stabilito.	S
TU04	Si verifica che l'ordine sia stato registrato correttamente in blockchain.	S
TU05	Si verifica il caso fallimentare in cui l'utente tenta di inviare un importo inferiore a quello richiesto.	S
TU06	Si verifica il caso fallimentare in cui l'utente prova a registrare un ordine che ha come venditore se stesso.	S
TU07	Si verifica il caso fallimentare in cui l'utente prova a inviare un ammontare pari a zero.	S
TU08	Si verifica il caso fallimentare in cui l'utente prova a inviare un ammontare negativo.	S
TU09	Si verifica il caso fallimentare in cui l'utente non ha abbastanza fondi.	S
TU10	Si verifica il caso fallimentare in cui il frontend richiede la creazione di un ordine con un id precedentemente usato.	S
TU11	Si verifica che lo stato dell'ordine sia modificato e settato su "CLOSED" una volta confermata la ricezione dall'acquirente.	S
TU12	Si verifica che il venditore abbia ricevuto i fondi correttamente una volta confermata la ricezione dall'acquirente.	S
TU13	Si verifica il caso fallimentare in cui l'utente tenta di confermare la ricezione di un ordine con id inesistente.	S
TU14	Si verifica il caso fallimentare in cui il codice di sblocco non coincide con quello registrato in blockchain.	S
TU15	Si verifica il caso fallimentare in cui l'ordine viene confermato da un utente con indirizzo differente da quello del compratore.	S
TU16	Si verifica che il rimborso sia richiesto dal compratore una volta settato su "CANCELLED" lo stato dell'ordine.	S

Continua nella pagina successiva

ID Test	Descrizione	Stato
TU17	Si verifica che il rimborso sia richiesto dal venditore una volta settato su "CANCELLED" lo stato dell'ordine.	S
TU18	Si verifica che i soldi siano restituiti correttamente al compratore quando è richiesto il rimborso dal compratore.	S
TU19	Si verifica che i soldi siano restituiti correttamente al compratore quando è richiesto il rimborso dal venditore.	S
TU20	Si verifica il caso fallimentare in cui non è possibile restituire l'ammontare di un ordine che non si trova nello stato "FILLED" (PAGATO).	NI
TU21	Si verifica il caso fallimentare in cui non è possibile chiedere il rimborso di un ordine chiuso.	NI
TU22	Si verifica il caso fallimentare in cui un utente con indirizzo diverso da quello del compratore o venditore richiede il rimborso.	NI
TU23	Si verifica che venga restituito correttamente il saldo del contratto.	S
TU24	Si verifica che venga restituito correttamente l'address dell'acquirente.	S
TU25	Si verifica che venga restituito correttamente l'address del venditore.	S
TU26	Si verifica che venga restituito correttamente l'ammontare da pagare.	S
TU27	Si verifica che venga restituito correttamente l'ammontare pagato.	S
TU28	Si verifica che venga restituito correttamente lo stato dell'ordine effettuato.	S
TU29	Si verifica che venga restituito correttamente l'id dell'ordine effettuato.	S
TU30	Si verifica che venga restituito correttamente l'ordine dell'acquirente.	S
TU31	Si verifica che venga restituito correttamente l'ordine del venditore.	S

Tabella 10: Test di unità - Solidity

4.4.1 Tracciamento test di unità - Solidity

ID Test	Metodo
TU01	smart-contract/test/ShopChain.test.js:getOrderCount()
TU02	smart-contract/test/ShopChain.test.js:newOrder(seller,ether_1,id1,from:buyer, value:ether_1)
TU03	smart-contract/test/ShopChain.test.js:equal(expectedBalance.toString(), newContractBalance.toString(),"contract is correctly filled")
TU04	smart-contract/test/ShopChain.test.js:getOrderById(id1)
TU05	smart-contract/test/ShopChain.test.js:newOrder(seller,ether_1,id2,from:buyer, value:ether_half)
TU06	smart-contract/test/ShopChain.test.js:newOrder(seller,ether_1,id2,from:seller, value:ether_1)
TU07	smart-contract/test/ShopChain.test.js:newOrder(seller,0,id2,from:seller, value:ether_1)
TU08	smart-contract/test/ShopChain.test.js:newOrder(seller,-1,id2,from:buyer, value:ether_1)
TU09	smart-contract/test/ShopChain.test.js:newOrder(seller,ether_big,id2,from:buyer, value:ether_big)
TU10	smart-contract/test/ShopChain.test.js:newOrder(seller,ether_1,id1,from:buyer, value:ether_1)
TU11	smart-contract/test/ShopChain.test.js:confirmReceived(id1,unlockCode,from:buyer)
TU12	smart-contract/test/ShopChain.test.js:confirmReceived(id1,unlockCode,from:buyer)
TU13	smart-contract/test/ShopChain.test.js:confirmReceived(id2,unlockCode,from:buyer)
TU14	smart-contract/test/ShopChain.test.js:confirmReceived(id1,12345,from:buyer)
TU15	smart-contract/test/ShopChain.test.js:confirmReceived(id1,unlockCode,from:seller)
TU16	smart-contract/test/ShopChain.test.js:refund(id1,from:buyer)
TU17	smart-contract/test/ShopChain.test.js:refund(id1,from:seller)
TU18	smart-contract/test/ShopChain.test.js:refund(id1,from:buyer)
TU19	smart-contract/test/ShopChain.test.js:refund(id1,from:seller)
TU23	smart-contract/test/ShopChain.test.js:contractBalance()

Continua nella pagina successiva

ID Test	Metodo
TU24	smart-contract/test/ShopChain.test.js:getOwnerAddress(id1)
TU25	smart-contract/test/ShopChain.test.js:getSellerAddress(id1)
TU26	smart-contract/test/ShopChain.test.js:getAmountToPay(id1)
TU27	smart-contract/test/ShopChain.test.js:getAmountPaid(id1)
TU28	smart-contract/test/ShopChain.test.js:getOrderState(id1)
TU29	smart-contract/test/ShopChain.test.js:getOrderById(id1)
TU30	smart-contract/test/ShopChain.test.js:getOrdersByBuyer(buyer)
TU31	smart-contract/test/ShopChain.test.js:getOrdersBySeller(seller)

Tabella 11: Tracciamento test di unità - metodi Solidity

A Resoconto attività di verifica

A.1 Verifica dei documenti

A.1.1 Indice di Gulpease

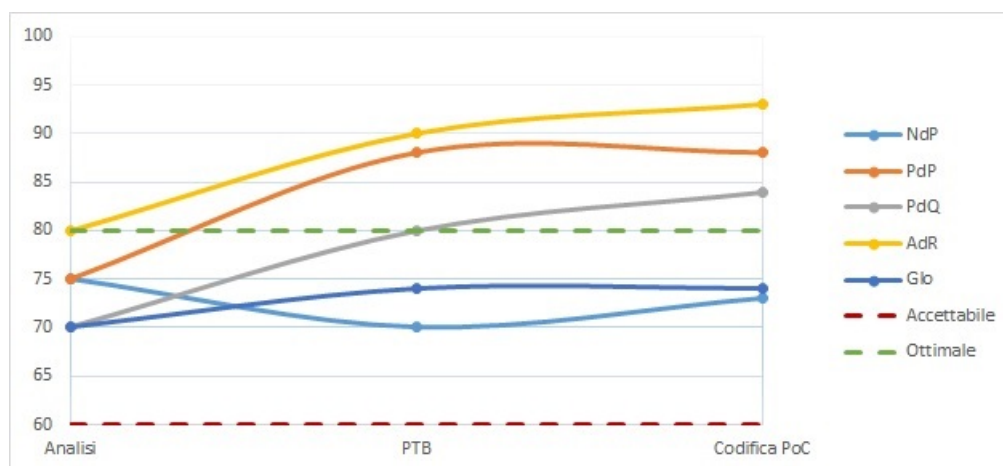


Figura 1: Indice di Gulpease di ciascun documento per periodo

A.1.2 Errori ortografici

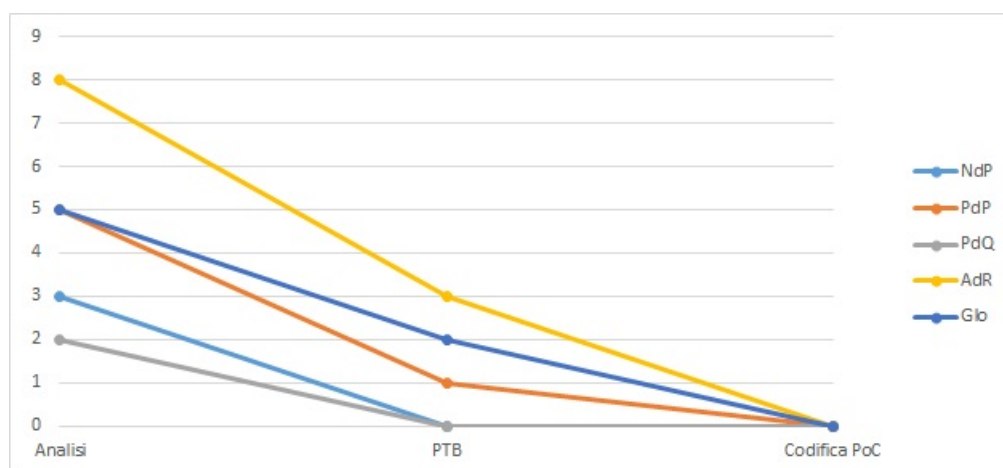


Figura 2: Errori ortografici individuati per periodo

A.2 Verifica dei processi

A.2.1 Estimated At Completion



Figura 3: Revisione del valore stimato per la realizzazione del progetto

A.2.2 Earned Value & Planned Value

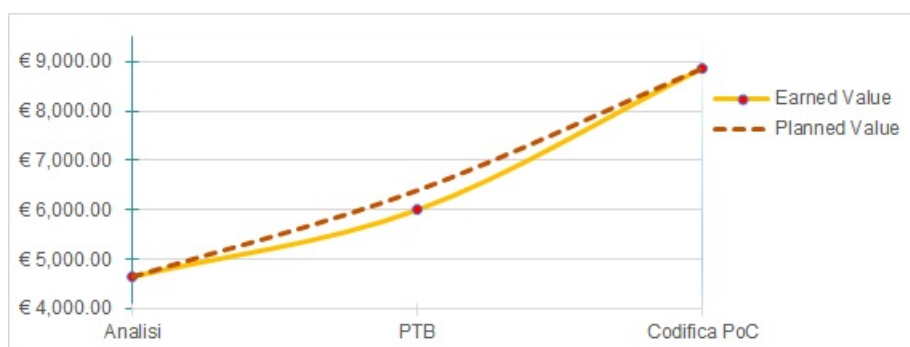


Figura 4: Valore delle attività realizzate e costo pianificato per realizzare le rimanenti

A.2.3 Actual Cost & Estimate To Complete

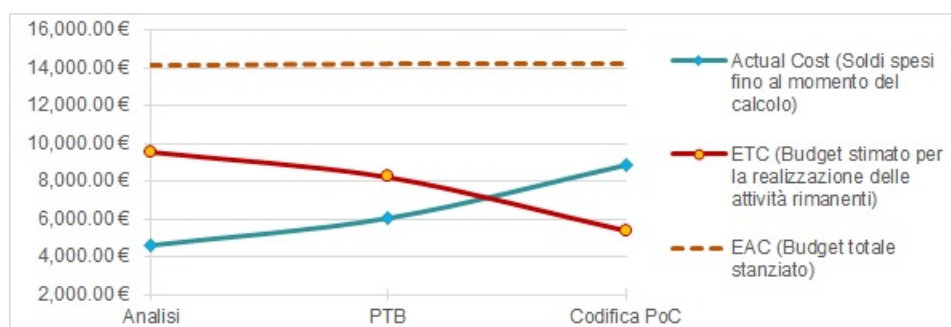


Figura 5: Costo effettivamente sostenuto e valore stimato per la realizzazione delle rimanenti attività

A.2.4 Schedule Variance & Cost Variance

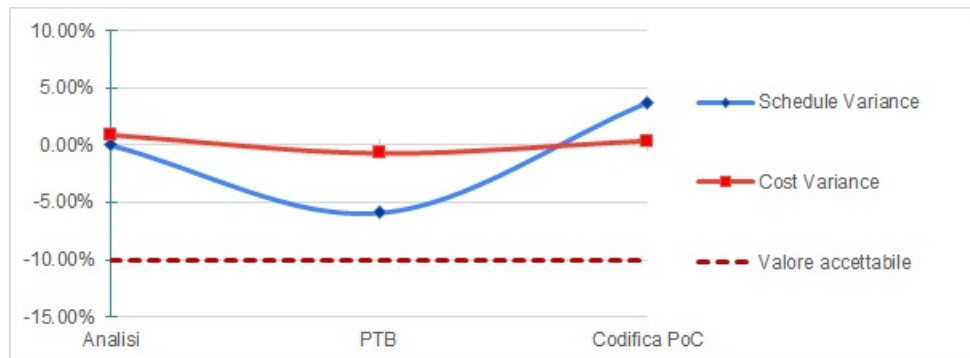


Figura 6: Cost Variance e Schedule Variance per periodo

A.2.5 Requirements Stability Index

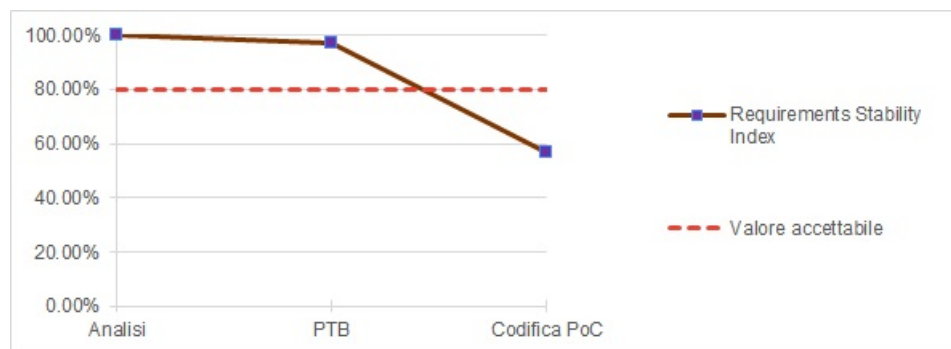


Figura 7: Variazione del numero di requisiti

A.2.6 Attualizzazione dei rischi



Figura 8: Rischi verificati per periodo

A.2.7 Metrics Satisfied



Figura 9: Percentuale di metriche soddisfatte per periodo

A.3 Verifica dei software

Il gruppo ha ritenuto più appropriato utilizzare come asse temporale, per le metriche relative il software, gli incrementi definiti nel *Piano di progetto*.

A.3.1 Facilità di utilizzo

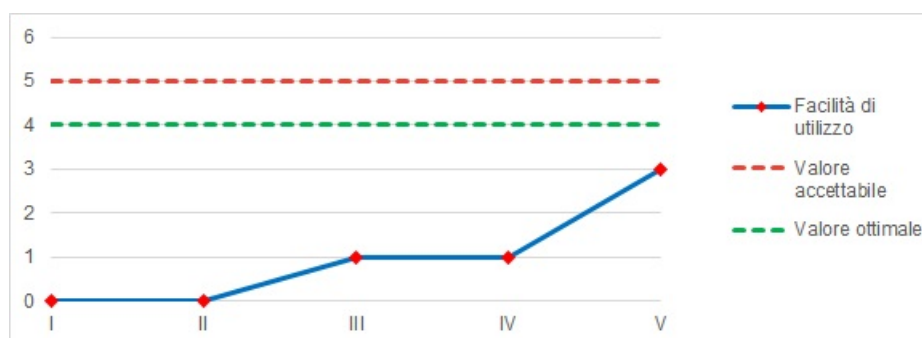


Figura 10: Quantitativo di click necessari per raggiungere la funzione desiderata

A.3.2 Versioni browser supportate

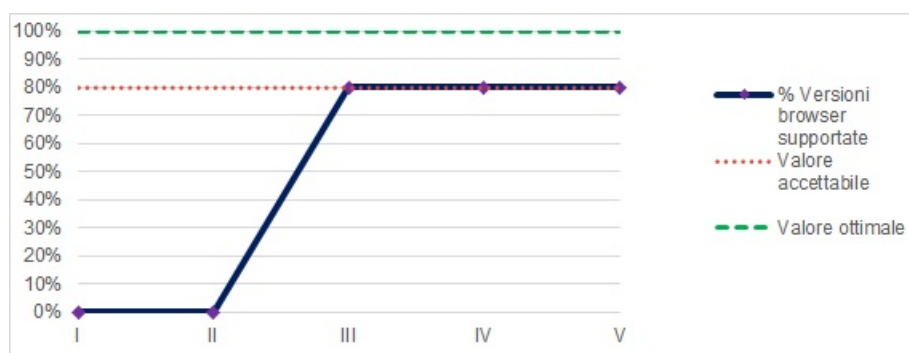


Figura 11: Versioni di browser supportate per periodo

A.3.3 Copertura requisiti obbligatori

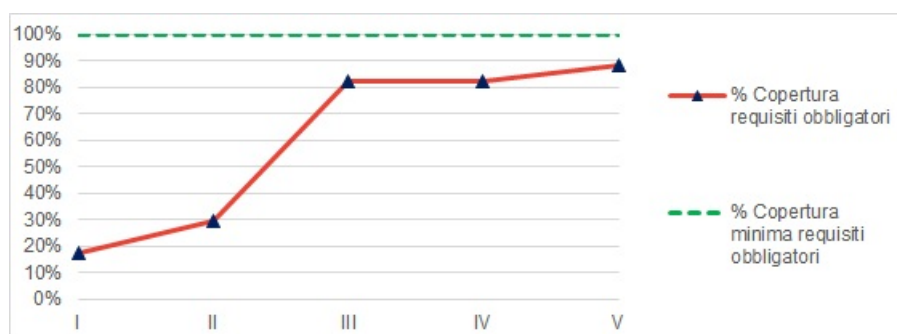


Figura 12: Percentuale di requisiti obbligatori implementati per periodo

A.3.4 Copertura requisiti desiderabili

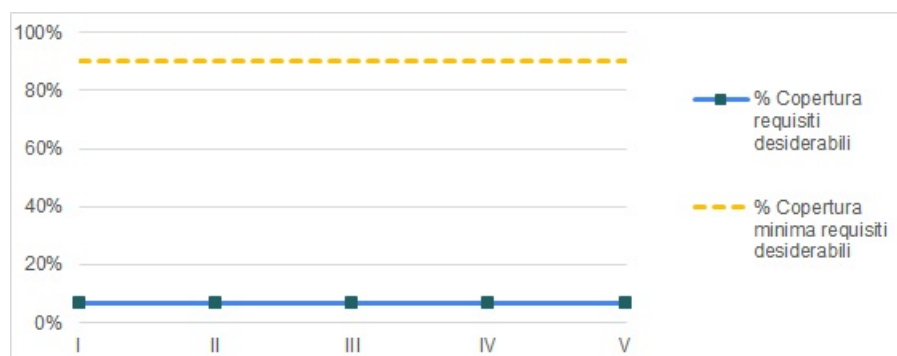


Figura 13: Percentuale di requisiti desiderabili implementati per periodo

A.3.5 Copertura requisiti opzionali

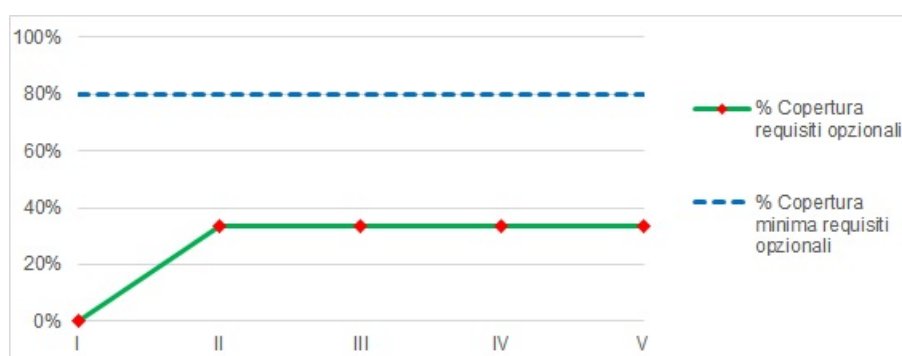


Figura 14: Percentuale di requisiti opzionali implementati per periodo

A.3.6 Solidity Statement Coverage

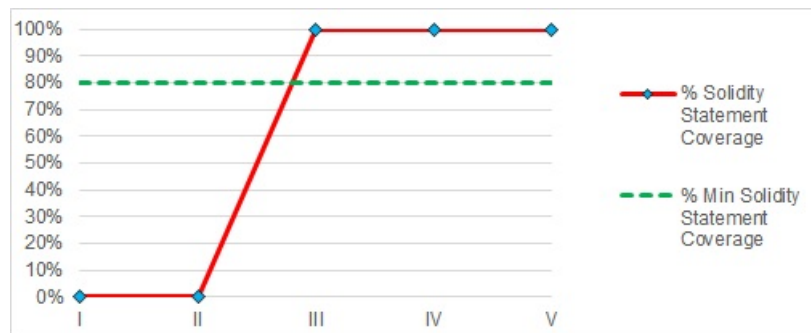


Figura 15: Statement Coverage riguardante il codice Solidity

A.3.7 Solidity Branch Coverage

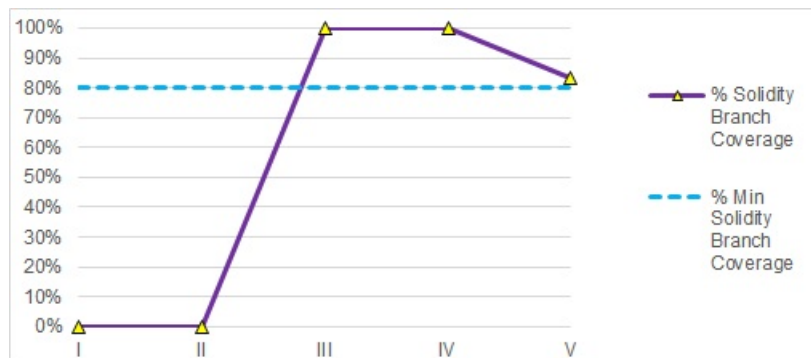


Figura 16: Branch Coverage riguardante il codice Solidity

A.3.8 Solidity Function Coverage

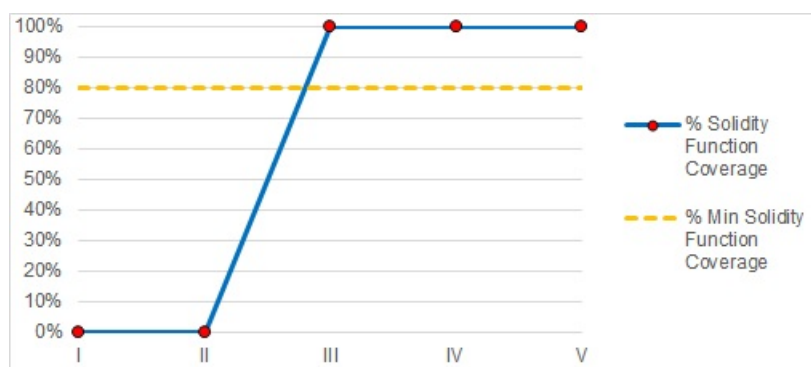


Figura 17: Function Coverage riguardante il codice Solidity

A.3.9 Solidity Line Coverage

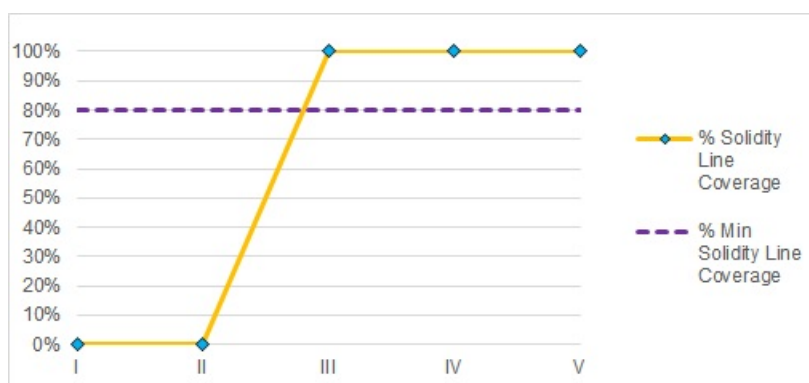


Figura 18: Line Coverage riguardante il codice Solidity