

# roundabout

*Roundabout - Etherless*

## Piano di Qualifica

<b>Versione</b>	0.0.6
<b>Approvazione</b>	
<b>Redazione</b>	Luca Benetazzo Nicoletta Fabro Feim Jakupi
<b>Verifica</b>	
<b>Stato</b>	Non approvato
<b>Uso</b>	Esterno
<b>Destinato a</b>	Roundabout Prof. Tullio Vardanega Prof. Riccardo Cardin

### **Descrizione**

*Questo documento descrive le operazioni di verifica e validazione seguite dal gruppo Roundabout per il progetto Etherless*

`team.roundabout.13@gmail.com`

## Registro delle modifiche

Versione	Data	Nominativo	Ruolo	Descrizione
0.0.6	2020-04-01	Luca Benetazzo	<i>Verificatore</i>	Stesura §C.
0.0.5	2020-04-01	Nicoletta Fabro	<i>Progettista</i>	Stesura §2.1, §2.2, §2.3.
0.0.4	2020-03-30	Luca Benetazzo	<i>Verificatore</i>	Stesura §B.
0.0.3	2020-03-26	Luca Benetazzo	<i>Verificatore</i>	Stesura §1.
0.0.2	2020-03-21	Nicoletta Fabro	<i>Progettista</i>	Organizzazione struttura documento.
0.0.1	2020-03-20	Luca Benetazzo	<i>Amministratore</i>	Creazione documento L <sup>A</sup> T <sub>E</sub> X.

## Indice

<b>1</b>	<b>Introduzione</b>	<b>5</b>
1.1	Premessa . . . . .	5
1.2	Scopo del documento . . . . .	5
1.3	Scopo del prodotto . . . . .	5
1.4	Glossario . . . . .	5
1.5	Riferimenti . . . . .	5
1.5.1	Riferimenti normativi . . . . .	5
1.5.2	Riferimenti informativi . . . . .	5
<b>2</b>	<b>Qualità di Processo</b>	<b>7</b>
2.1	Scopo . . . . .	7
2.2	Processi Primari . . . . .	7
2.2.1	Processi di sviluppo . . . . .	7
2.2.1.1	Analisi dei Requisiti . . . . .	7
2.2.1.2	Progettazione . . . . .	7
2.2.1.3	Codifica . . . . .	8
2.3	Processi di Supporto . . . . .	9
2.3.0.1	Pianificazione . . . . .	9
2.3.0.2	Verifica . . . . .	11
2.3.0.3	Documentazione . . . . .	11
2.4	Processi Organizzativi . . . . .	11
2.4.0.1	Gestione della Qualità . . . . .	11
<b>3</b>	<b>Qualità di Prodotto</b>	<b>13</b>
3.1	Scopo . . . . .	13
3.2	Funzionalità . . . . .	13
3.3	Affidabilità . . . . .	13
3.4	Usabilità . . . . .	13
3.5	Manutenibilità . . . . .	13
<b>4</b>	<b>Specifica dei test</b>	<b>14</b>
4.1	Scopo . . . . .	14
4.2	Test di accettazione . . . . .	14
<b>A</b>	<b>Standard di qualità</b>	<b>15</b>
A.1	ISO/IEC 9126 . . . . .	15
A.1.1	Modello per la qualità del software . . . . .	15
A.2	ISO/IEC 15504 . . . . .	16
A.3	Ciclo di Deming . . . . .	16
<b>B</b>	<b>Valutazioni per il miglioramento</b>	<b>17</b>
B.1	Valutazioni sull'organizzazione . . . . .	17
B.2	Valutazioni sui ruoli . . . . .	17
B.3	Valutazioni sugli strumenti di lavoro . . . . .	18

<b>C</b>	<b>Resoconto delle attività di verifica</b>	<b>19</b>
C.1	Verifiche statiche . . . . .	19
C.2	Verifiche requisiti . . . . .	19
C.3	Verifiche automatizzate . . . . .	19

## Elenco delle figure

A.1.1Modello ISO/IEC 9126 . . . . .	16
-------------------------------------	----

# 1 Introduzione

## 1.1 Premessa

Il *Piano di Qualifica* è un documento di cui si prevede la stesura durante l'intera durata del progetto, adottando una modalità incrementale. Per questo motivo, non è da considerarsi equivalente ad un documento completo.

## 1.2 Scopo del documento

Questo documento contiene tutte le strategie di verifica e validazione adottate dal gruppo *Roundabout*, al fine di garantire la qualità di prodotto e processo. Per ottenere questo obiettivo viene applicato una verifica continua sui processi in corso e sulle attività svolte. Procedendo in questo modo si eviteranno più facilmente eventuali malformità e si consentirà una manutenzione qualitativamente migliore.

## 1.3 Scopo del prodotto

L'applicativo che si vuole sviluppare è *Etherless*, una piattaforma cloud che sfrutta la tecnologia degli smart contract caratteristica del network Ethereum. Lo scopo di *Etherless* è duplice: da una parte permettere agli sviluppatori *software* di rilasciare funzioni Javascript nel cloud, dall'altra permettere agli *utenti* di beneficiare di queste funzioni in seguito ad un pagamento per il loro uso. *Etherless* è gestita e mantenuta dai suoi *amministratori*.

## 1.4 Glossario

Al fine di evitare possibili ambiguità, i termini tecnici utilizzati nei documenti formali vengono chiariti ed approfonditi nel *Glossario Interno 1.0.0*. Per facilitare la lettura, i termini presenti in tale documento sono contrassegnati in tutto il resto della documentazione da una 'G' a pedice.

## 1.5 Riferimenti

### 1.5.1 Riferimenti normativi

- **Norme di Progetto:** *Norme di Progetto v1.0.0*;
- **Capitolato d'appalto C2 - Etherless:**  
<https://www.math.unipd.it/~tullio/IS-1/2019/Progetto/C2.pdf>.

### 1.5.2 Riferimenti informativi

- **Standard ISO/IEC 9126:**  
[https://it.wikipedia.org/wiki/ISO/IEC\\_9126](https://it.wikipedia.org/wiki/ISO/IEC_9126);
- **Standard ISO/IEC 15504:**  
[https://en.wikipedia.org/wiki/ISO/IEC\\_15504](https://en.wikipedia.org/wiki/ISO/IEC_15504);
- **Ciclo di Deming:**  
[https://it.wikipedia.org/wiki/Ciclo\\_di\\_Deming](https://it.wikipedia.org/wiki/Ciclo_di_Deming);
- **Indice di Gulpease:**  
[https://it.wikipedia.org/wiki/Indice\\_Gulpease](https://it.wikipedia.org/wiki/Indice_Gulpease);

- **Slide Qualità di prodotto:**  
<https://www.math.unipd.it/~tullio/IS-1/2019/Dispense/L12.pdf>;
- **Slide Qualità di processo:**  
<https://www.math.unipd.it/~tullio/IS-1/2019/Dispense/L13.pdf>;
- **Slide Verifica e Validazione:**
  - <https://www.math.unipd.it/~tullio/IS-1/2019/Dispense/L14.pdf>;
  - <https://www.math.unipd.it/~tullio/IS-1/2019/Dispense/L15.pdf>;
  - <https://www.math.unipd.it/~tullio/IS-1/2019/Dispense/L16.pdf>;

## 2 Qualità di Processo

### 2.1 Scopo

Lo scopo della seguente sezione è quello di illustrare le metriche adottate per garantire qualità nello svolgimento dei processi, precedentemente scelti tra quelli proposti dallo standard ISO/IEC 12207:1995 ed illustrati nel documento Norme di Progetto.

### 2.2 Processi Primari

#### 2.2.1 Processi di sviluppo

##### 2.2.1.1 Analisi dei Requisiti

Le metriche usate per l'attività di analisi sono le seguenti:

##### Percentuale dei Requisiti Obbligatori Soddisfatti (PROS)

E' un valore percentuale che indica la quantità di requisiti obbligatori adempiti nel corso del progetto.

- Misurazione:  $PROS = \frac{\text{requisiti obbligatori soddisfatti}}{\text{requisiti obbligatori totali}}$ ;
- Valore minimo accettabile: 100%;
- Valore preferibile: 100%.

##### 2.2.1.2 Progettazione

Le metriche usate per l'attività di progettazione sono le seguenti:

##### Structural Fan-In (SFIN)

E' un valore intero che indica quante componenti utilizzano un dato modulo. Uno *SFIN* alto sta a significare un consistente riuso della componente.

- Misurazione: valore intero che indica il conteggio delle componenti;
- Valore minimo accettabile:  $\geq 0$ ;
- Valore preferibile:  $\geq 1$ .

##### Structural Fan-Out (SFOUT)

E' un valore intero che indica quante componenti vengono utilizzate dalla componente in questione. Uno *SFOUT* alto sta a significare un significativo accoppiamento della componente.

- Misurazione: valore intero che indica il conteggio delle componenti;
- Valore minimo accettabile:  $= 0$ ;
- Valore preferibile:  $\leq 6$ .



### 2.2.1.3 Codifica

Le metriche usate per l'attività di codifica sono le seguenti:

#### Coupling Between Objects (CBO)

E' un valore intero che indica il grado di accoppiamento tra le classi di oggetti. Una classe A si dice *accoppiata* ad una classe B se A usa metodi o variabili definite in B.

- Misurazione: valore intero;
- Valore minimo accettabile:  $0 \leq \text{CBO} \leq 6$ ;
- Valore preferibile:  $0 \leq \text{CBO} \leq 1$ .

#### Complessità ciclomatica

E' una metrica utilizzata per stimare la complessità di funzioni, moduli, metodi o classi su un programma. Ciò viene fatto mediante la determinazione del numero dei cammini linearmente indipendenti attraverso il grafo di controllo di flusso. Un valore troppo elevato indica un'eccessiva complessità del codice che rende conseguentemente l'attività di manutenzione più difficile. Allo stesso tempo, un valore eccessivamente basso potrebbe essere indicatore di una scarsa efficienza dei metodi.

- Misurazione: valore intero;
- Valore minimo accettabile:  $1 \leq \text{complessità ciclomatica} \leq 15$ ;
- Valore preferibile:  $1 \leq \text{complessità ciclomatica} \leq 10$ .

#### Rapporto linee di codice per linee di commento

Indica il rapporto tra le righe di codice e le righe di commento ad esso corrispondenti (escludendo le righe vuote), al fine di stimare il livello di difficoltà di manutenibilità del codice. Un rapporto troppo basso, infatti, potrebbe essere indicatore di scarse informazioni necessarie alla comprensione del codice scritto.

- Misurazione: valore decimale;
- Valore minimo accettabile:  $\frac{\text{linee di codice}}{\text{linee di commento}} \geq 0.25$ ;
- Valore preferibile: rapporto  $\frac{\text{linee di codice}}{\text{linee di commento}} \geq 0.30$ .

#### Livello di annidamento

E' un valore intero che indica il livello di annidamento nei vari metodi tenendo conto della presenza di strutture di controllo annidate: un alto livello di annidamento risulta in codice complesso e di difficile manutenzione.

- Misurazione: valore intero;
- Valore minimo accettabile:  $1 \leq \text{livello annidamento} \leq 7$ ;
- Valore preferibile:  $1 \leq \text{livello annidamento} \leq 3$ .

### Numero di parametri per metodo

E' un numero intero che indica il numero di parametri di un metodo. Un valore troppo elevato denota un grado di complessità eccessivamente elevato del metodo

- Misurazione: valore intero;
- Valore minimo accettabile:  $0 \leq \text{numero totale attributi} \leq 8$ ;
- Valore preferibile:  $0 \leq \text{numero totale attributi} \leq 4$ .

### Numero di attributi per classe

E' un numero intero che indica il numero totale di attributi presenti all'interno di una classe. Un valore troppo elevato denota un carico di responsabilità eccessiva per la classe, suggerendo di scomporre la stessa seguendo principi quali il *Single Responsibility Principle*.

- Misurazione: valore intero;
- Valore minimo accettabile:  $1 \leq \text{numero totale attributi} \leq 15$ ;
- Valore preferibile:  $1 \leq \text{numero totale attributi} \leq 8$ .

## 2.3 Processi di Supporto

### 2.3.0.1 Pianificazione

Le metriche usate per l'attività di pianificazione sono le seguenti:

#### Budget at Completion (BAC)

Equivale al budget totale allocato per il progetto.

- Misurazione: numero intero;
- Valore minimo accettabile: valore del preventivo con un errore massimo del 5%, ovvero  $\text{preventivo}-5\% \leq BAC \leq \text{preventivo}+5\%$ ;
- Valore preferibile: pari al preventivo.

#### Cost Variance (CV)

Detta anche *Budget Variance*, in quanto misura il budget a disposizione nel corso di un progetto software. In particolare, la *Cost Variance* è la differenza tra *Earned Value* e *Actual Cost*, ovvero tra ciò che si aveva pianificato di spendere e ciò che si è effettivamente speso nel corso del progetto. Se la *Cost Variance* ha valore negativo significa che si è *over budget*, se è nulla si è *on budget*, mentre se è positiva si è *under budget*.

- Misurazione:  $CV = EV - AC$ ;
- Valore minimo accettabile: 0;
- Valore preferibile:  $> 0$ ;

**Schedule Variance (SV)**

La *Schedule Variance* è una misura che si usa per indicare lo stato di avanzamento di un progetto software e viene calcolata mediante la differenza tra *Earned Value* e *Planned Value*. Se la *Schedule Variance* ha valore negativo significa che lo stato di avanzamento del progetto è in ritardo rispetto alla pianificazione, se è nulla lo stato di avanzamento del progetto è nei tempi previsti, mentre se è positiva significa che si è in anticipo rispetto alla pianificazione.

- Misurazione:  $SV = EV - PV$ ;
- Valore minimo accettabile: 0;
- Valore preferibile:  $> 0$ .

**Earned Value (EV)**

Anche chiamata *Budgeted Cost of Work Performed (BCWP)* indica la quantità di lavoro compiuta al momento del calcolo. Viene calcolata dal budget di progetto come segue:

- Misurazione:  $EV = \% \text{ di lavoro completato} \cdot BAC$ ;
- Valore minimo accettabile:  $\geq 0$ ;
- Valore preferibile:  $\geq 0$ ;

**Actual Cost (AC)**

Anche conosciuto come *Actual Cost of Work Performed (ACWP)* indica la quantità di budget spesa al momento del calcolo.

- Misurazione: numero intero;
- Valore minimo accettabile:  $0 \leq AC < BAC$ ;
- Valore preferibile:  $0 \leq AC < PV$ .

**Planned Value (PV)**

Corrisponde al valore del lavoro pianificato al momento del calcolo, ovvero al denaro che si dovrebbe aver guadagnato fino a quel momento.

- Misurazione:  $\% \text{ di lavoro pianificato} \cdot BAC$ ;
- Valore minimo accettabile:  $\geq 0$ ;
- Valore preferibile:  $\geq 0$ .

**Correlazione tra CV e SV**

Lo stato di un progetto è esprimibile dalla correlazione tra *Cost Variance* e *Schedule Variance*, in particolare:

1. **SV e CV positive:** il progetto è in anticipo rispetto alla pianificazione e rientra nel budget previsto;

2. **SV positiva, CV negativa:** il progetto è in anticipo rispetto alla pianificazione ma ha superato il budget allocato;
3. **SV negativa, CV positiva:** il progetto è in ritardo rispetto alla pianificazione ma rientra nel budget previsto;
4. **SV e CV negative:** il progetto è in ritardo rispetto alla pianificazione e ha superato il budget previsto.

### 2.3.0.2 Verifica

Le metriche usate per l'attività di verifica sono le seguenti:

#### Code Coverage

E' la percentuale di linee di codice che sono state eseguite dai test dopo un'esecuzione.

- Misurazione:  $CC = \frac{\text{linee di codice eseguite dal test}}{\text{linee di codice totali}}$ ;
- Valore minimo accettabile: 80%;
- Valore preferibile: 100%.

### 2.3.0.3 Documentazione

Le metriche usate per l'attività di documentazione sono le seguenti:

#### Indice di Gulpease

E' un indice che valuta la leggibilità del testo. E' tarato sulla lingua italiana e considera la lunghezza delle parole, il numero delle frasi ed il numero delle parole totali. Il valore risultante è compreso tra 0 e 100, dove un valore di indice più alto corrisponde ad un indice di leggibilità più semplice.

- Misurazione:  $Gulpease = 89 + \frac{300 \cdot (\text{numero delle frasi}) - 10 \cdot (\text{numero delle lettere})}{\text{numero delle parole}}$ ;
- Valore minimo accettabile:  $\geq 40$ ;
- Valore preferibile:  $\geq 60$ .

#### Correttezza ortografica

Tutti i documenti devono essere privi di errori grammaticali od ortografici.

- Misurazione: numero intero che indica il numero di errori presenti nel testo;
- Valore minimo accettabile: 0;
- Valore preferibile: 0.

## 2.4 Processi Organizzativi

### 2.4.0.1 Gestione della Qualità

Le metriche usate per la gestione della qualità sono le seguenti:

**Percentuale di metriche soddisfatte (PMS)**

E' un valore percentuale che indica quante metriche raggiungono soglie accettabili sul totale delle metriche considerate. Un valore percentuale basso potrebbe essere indicatore di scarsa qualità, incorrettezza di calcolo o scelta metriche poco adeguate.

- $PMS = \frac{\text{numero di metriche soddisfatte}}{\text{numero totale di metriche}}$
- Valore minimo accettabile: 60%;
- Valore preferibile: 90%.

### **3 Qualità di Prodotto**

#### **3.1 Scopo**

#### **3.2 Funzionalità**

#### **3.3 Affidabilità**

#### **3.4 Usabilità**

#### **3.5 Manutenibilità**

## **4 Specifica dei test**

### **4.1 Scopo**

### **4.2 Test di accettazione**

## A Standard di qualità

### A.1 ISO/IEC 9126

ISO/IEC 9126 è lo standard generale usato per valutare la qualità del software. Esso è articolato in quattro parti:

- Modello per la qualità del software
- Metriche per la qualità interna
- Metriche per la qualità esterna
- Metriche per la qualità in uso

#### A.1.1 Modello per la qualità del software

Il modello di qualità stabilito nella prima parte dello standard ISO/IEC 9126 è composto da sei caratteristiche generali e varie sotto caratteristiche misurabili attraverso delle metriche. Tali caratteristiche sono:

##### Funzionalità

E' la capacità di un prodotto software di fornire funzioni che soddisfano esigenze stabilite nell'Analisi dei Requisiti e che permettono di operare in condizioni specifiche. Questa capacità si traduce nelle seguenti caratteristiche:

- **Appropriatezza:** capacità di fornire funzioni appropriate per attività specifiche che permettano di raggiungere gli obiettivi prefissati;
- **Accuratezza:** capacità di fornire risultati corretti e con la precisione richiesta;
- **Interoperabilità:** capacità di interagire con uno o più sistemi specificati;
- **Conformità:** capacità di aderire a standard;
- **Sicurezza:** capacità di proteggere informazioni e dati.

##### Affidabilità

E' la capacità del prodotto software di mantenere uno dato livello di prestazioni quando usato in specifiche condizioni. Questa capacità si traduce nelle seguenti caratteristiche:

- **Maturità:** capacità di evitare il verificarsi di errori o malfunzionamenti in fase di esecuzione;
- **Tolleranza agli errori:** capacità di mantenere livelli predeterminati di prestazioni anche in presenza di malfunzionamenti o errori;
- **Recuperabilità:** capacità di ripristinare il livello di prestazioni e di recupero delle informazioni rilevanti in seguito ad un malfunzionamento;
- **Aderenza:** capacità di aderire a standard e regole inerenti all'affidabilità.



**Efficienza**

E' la capacità di eseguire i compiti prefissati minimizzando il tempo necessario e sfruttando al meglio le risorse disponibili. Questa capacità si traduce nelle seguenti caratteristiche:

- **Nel tempo:**
- **Nello spazio:**

**Usabilità**

Questa capacità si traduce nelle seguenti caratteristiche:

- **Comprensibilità:**
- **Apprendibilità:**
- **Operabilità:**
- **Attrattività:**
- **Conformità:**

**Manutenibilità**

Questa capacità si traduce nelle seguenti caratteristiche:

- **Analizzabilità:**
- **Modificabilità:**
- **Stabilità:**
- **Testabilità:**

**Portabilità**

Questa capacità si traduce nelle seguenti caratteristiche:

- **Adattabilità:**
- **Installabilità:**
- **Conformità:**
- **Sostituibilità:**

A.1.2 Metriche per la qualità interna

A.1.3 Metriche per la qualità esterna

A.1.4 Metrica per la qualità in uso

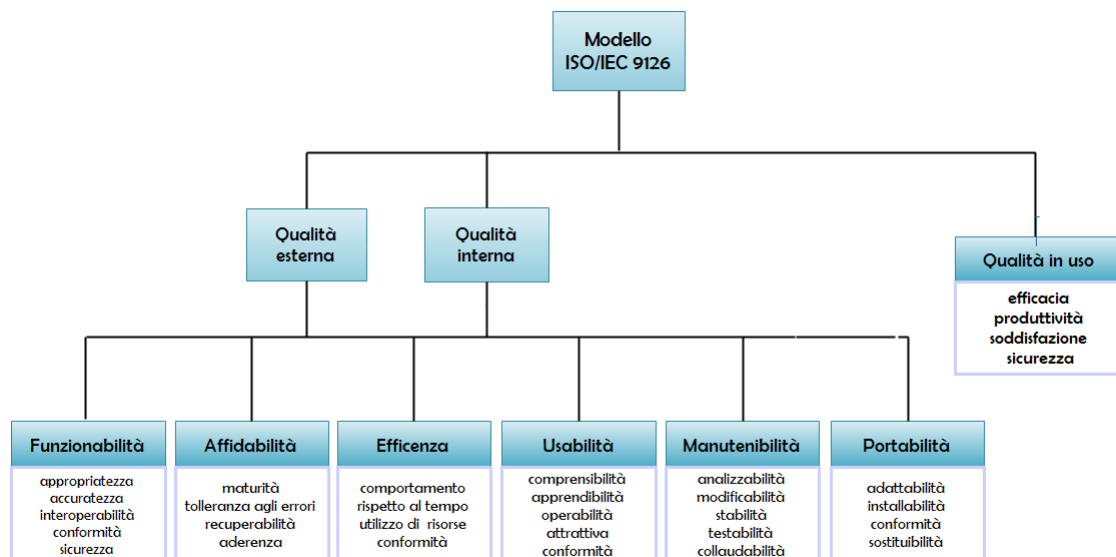


Figura A.1.1: Modello ISO/IEC 9126 (fonte: Wikipedia)

A.2 ISO/IEC 15504

A.3 Ciclo di Deming

## B Valutazioni per il miglioramento

Questa sezione riporta i problemi riscontrati durante il corso del progetto dal gruppo *Roundabout*. Ogni problema viene valutato per trovare una possibile soluzione e quindi un miglioramento il più efficace ed efficiente possibile.

Si espongono di seguito i problemi incontrati divisi in 3 raggruppamenti:

- **organizzazione:** problemi relativi l'organizzazione e la comunicazione all'interno del gruppo;
- **ruoli:** problemi relativi allo svolgimento dei diversi ruoli;
- **strumenti di lavoro:** problemi relativi l'uso degli strumenti utilizzati.

### B.1 Valutazioni sull'organizzazione

Tabella B.1.1: Valutazioni Organizzazione

Problema	Soluzione
<b>Riunioni Interne:</b> si è rivelato un problema organizzativo l'impossibilità di vedersi fisicamente a causa della situazione di emergenza COVID-19	Abbiamo concordato di utilizzare maggiormente strumenti di collaborazione che consentono, oltre alla possibilità di effettuare videochiamate, una comunicazione semplificata per i diversi problemi che si possono verificare.
<b>Appuntamenti:</b> Problema a definire una calendarizzazione degli incontri tra i vari membri del gruppo	Abbiamo definito che le riunioni interne saranno effettuate cadenzialmente due volte alla settimana il martedì e il venerdì, salvo esigenze particolari.
<b>Riunioni Esterne:</b> Durante la prima riunione effettuata con il <i>Proponente</i> a mezzo Skype, si è valutato il problema comune di connessione instabile e conseguente perdita di parole durante la conversazione.	Risolto proponendo al <i>Proponente</i> incontri telematici su piattaforma Zoom, molto più leggera e con limitati problemi di chiamata.

### B.2 Valutazioni sui ruoli

Tabella B.2.1: Valutazioni Ruoli

Problema	Soluzione
<b>Rivestire un ruolo:</b> Il problema comune a tutti i ruoli è stato quello di doversi adattare ad una mentalità diversa in base al contesto richiesto, considerato il vincolo che ogni membro dovrà ricoprire un ruolo descritto nelle <i>Norme di Progetto</i> .	Valutato che il maggior impatto di questa problematica si verifica nella fase iniziale di ogni "cambio ruolo", si è deciso di limitare le rotazioni indicativamente ogni due settimane cercando di non lasciare lavori in sospeso al membro successivo. In ogni caso vige il buon senso e la collaborazione reciproca.
<b>Responsabile di Progetto:</b>	
<b>Amministratore:</b>	
<b>Analista:</b>	
<b>Verificatore:</b>	

### B.3 Valutazioni sugli strumenti di lavoro

Tabella B.3.1: Valutazioni Strumenti di Lavoro

Problema	Soluzione
<b>L<sup>A</sup>T<sub>E</sub>X:</b> si è rivelato un problema l'utilizzo di questo strumento, in quanto la maggior parte del gruppo <i>Roundabout</i> non lo aveva mai utilizzato prima.	La soluzione è stata quella di usufruire dell'esperienza maturata da parte di alcuni membri del gruppo per apprendere le basi di utilizzo: prima creando un template standar, poi illustrandolo assieme ad alcuni comandi che avremmo utilizzato con maggiore frequenza.
<b>Ethereum:</b> si è rivelato un problema la non conoscenza di questa piattaforma	Si è colmata questa mancanza tramite ricerca personale e studio autonomo.
<b>Omogeneità dei documenti prodotti:</b> Considerato che la stesura di un documento può essere effettuata anche da più persone che ricoprono lo stesso ruolo in contemporanea, si è verificato il problema di omogeneità all'interno dei documenti	La soluzione migliore è stata quella di concordare assieme nelle <i>Norme di Progetto</i> gli utilizzi di maiuscole, minuscole, corsivo, grassetto, etc.

## C Resoconto delle attività di verifica

### C.1 Verifiche statiche

Ogni documento prodotto è stato analizzato da parte dei *Verificatori*, adottando un metodo Walkthrough<sub>G</sub> ed Inspection<sub>G</sub>.

Terminata questa analisi, in accordo con il redattore, si procede alla risoluzione di lacune eventualmente presenti.

### C.2 Verifiche requisiti

Questo tipo di verifica è necessario per accertarsi che, la relazione tra casi d'uso, requisiti e fonti non abbia discrepanze. Per facilitare questa verifica si è scelto di utilizzare il software PragmaDB.

### C.3 Verifiche automatizzate

Nella seguente tabella vengono riportati i valori di Gulpease<sub>G</sub> calcolati per ogni documento. I calcoli sono stati effettuati escludendo le intestazioni e le note a piè di pagina, in modo da avere un risultato valido ed attendibile. L'esito della verifica è da intendersi *Positivo* qualora l'indice di Gulpease abbia valore maggiore di 40.

Tabella C.3.1: Verifica Gulpease documenti

Documento	Gulpease	Esito
Analisi dei Requisiti v1.0.0	0	Non Positivo
Glossario v1.0.0	0	Non Positivo
Norme di Progetto v1.0.0	0	Non Positivo
Piano di Progetto v1.0.0	0	Non Positivo
Studio di Fattibilità v1.0.0	0	Non Positivo
Verbale xyz v1.0.0	0	Non Positivo
Verbale xyz v1.0.0	0	Non Positivo
Verbale xyz v1.0.0	0	Non Positivo
Verbale xyz v1.0.0	0	Non Positivo
Verbale xyz v1.0.0	0	Non Positivo